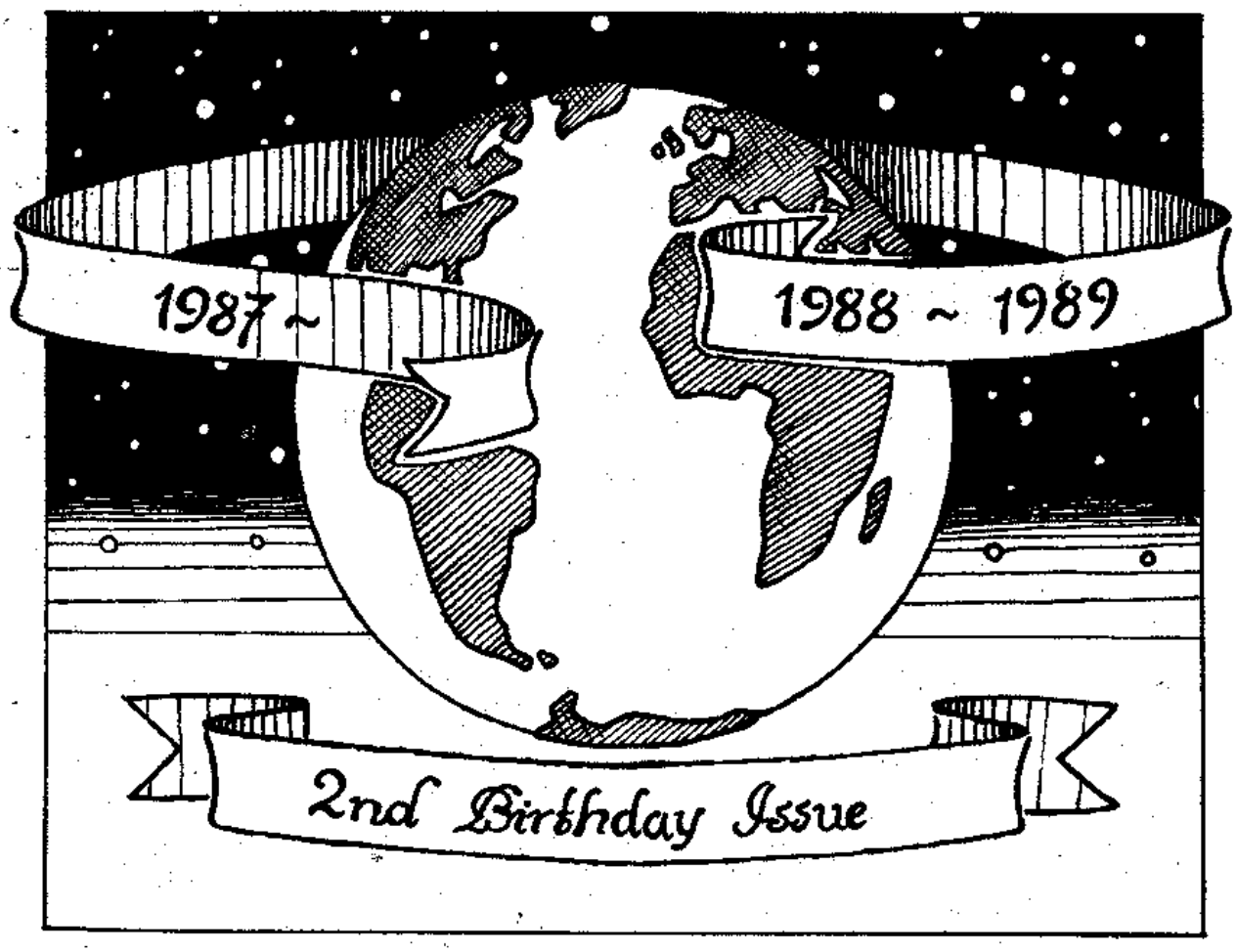
Vol 3 – No 1. September 1989.

FORMAT

FOR SPECTRUM USERS



THE BIG ONE

BUSINESS AND PLEASURE

The MGT Lifetime Drive. A Citizen 1Mbyte 3½" drive with its own power supply. Works with your DISCiPLE or PLUS D. Also works with your Q.L., Amiga, ST, BBC or even PC! All you change is the cable and switches. A lifetime of business and pleasure!

Lifetime Drive.....**£129.95** inc VAT
Spectrum/BBC Cable.....**£10.00** inc VAT
All other cables (each).....**£18.50** inc VAT
Shipping charge.....**£4.50**

MGT
MILES GORDON TECHNOLOGY

Miles Gordon Technology plc

Lakeside, Phoenix Way,
Swansea Enterprise Park,
Swansea, SA7 9EH ☎ 0792 791100

CONTENTS

FORMAT is published by INDUG. 34 Bourton Road, Gloucester, GL4 OLE, England. Telephone 0452-412572. DISCiPLE and PLUS D are trade marks of MILES GORDON TECHNOLOGY plc. Lakeside Technology Park, Phoenix Way, Swansea, South Wales, SA7 9EH. Telephone 0792-791100.

Printed by D.S.LITHO. Gloucester. Tel:- (0452) 23198.

NEWS ON 4

LIFETIME ADVANCES.

MGT have made a number of advances in marketing their successful LIFETIME disc drives. First a number of distributors are being appointed to ensure a wider availability. This push has been spearheaded by MGT's new Group Sales Manager, Richard Millar (formally with Citizen UK). The first of these new distributors to be appointed is Hugh Symons Distribution of Poole.

Increased volume sales have now allowed MGT to give Lifetime Drive customers their first connection cable FREE. This represents an average saving of about 10%.

SCOTTISH MICRO SHOW.

A Scottish Micro Show will be held on the 18th November at The Forum, Almondvale West, Livinston. The venue is just 14 miles from Edinburgh, 25 miles from Glasgow and only 3 miles of the M8 motorway (junction 3). There will be plenty of FREE parking for visitors (unlike London shows).

The show is being organized by B&H Computers of Halifax (tel: 0422 54581) and will be open from 10am to 5pm.

GREMLIN MOVES.

Long standing software company GREMLIN, one of the few survivors from the hay-day of home computing, is moving back to Sheffield. The move follows a buy-out of the company, by boss Ian Stewart, from US Gold.

BT TAKES OVER MICRONET.

British Telecom has taken full control of the Telemap Group who control the Micronet system. BT did have a 40% holding in Telemap but now have 100% of the shares. BT spokesman said there would be no change in Telemap's operations which, apart from Micronet, include Shades and InterBusiness.

MODEM MAKERS UP FOR SALE.

Miracom, makers of a wide range of computer modem equipment, have called in the receivers. Cork Gully (a well respected firm of accountants) have been appointed as Administrative Receivers to keep the company trading while a buyer is sought. US Robotics who supply Miracom with much of their product, was at one time interested in buying the company but pulled out.

ANTI-PIRACY WARNING.

FAST, the Federation Against Software Theft, are making representations to software companies to get standard warning notices, against software copying, put on the packaging and title screens of all programs. An agreed legal wording would allow FAST to proceed against software pirates with more effect. Most software companies put some form of notice on packaging, but the wording varies so much that it is difficult for legal cases to be brought against pirates except by an individual company.

+2 CASSETTE INTERFACE

Built-in cassette unit on your +2 gone wrong? Well J & P Electronics may have the answer. For £19.95 (far cheaper than a new internal unit) they will sell you a plug-in port that allows you to use an external cassette recorder. J & P can be contacted on Kidderminster (0562) 753893.

If you have any news items you want to pass on then send them in. Please mark the envelope NEWS in the top corner.

A very special editorial for a very special issue, Yes its BIRTHDAY time again. Put on your party hats, crack open the champagne, FORMAT has reached its second birthday. Thanks to all of you who sent letters of congratulation, its nice to know FORMAT is held in such high esteem. In this special editorial I want to look back over the history of FORMAT/INDUG and forward at its future.

But before I get stuck-in, let me give you the latest news on the 1989 Subscription drive I launched last month. The first form landed on my desk less than a week after FORMAT was posted out, not bad at all. Even more encouraging has been the number of you who have been asking for more copies of the form (remember though, you can photo-copy it as many times as you like to pass on to your friends). However I have had the usual moan from some people "I don't know anyone else with a Spectrum". Rubbish. And I'll say it again RUBBISH. Many Spectrum users don't go round shouting at the top of their voices "I'm a Spectrum user" (unlike Amiga and ST owners - don't they just bore the pants of you?). But there are more Spectrum owners than Commodore, BBC and Amstrad owners combined.

So start talking to your work-mates or school-chums, you will soon find that the Spectrum owners come out of the closet. Tell them how much they are missing out on, if they don't get FORMAT each month. Get them to send off their subscription today. Spread the word about FORMAT and reap the reward. Remember for each new subscriber I add 50p to the pot. For each new reader you introduce you get a chance to win that pot when the draw takes place in January. Don't take a back seat, help build FORMAT's readership, it benefits you in the long run.

Right, lets get back to my birthday essay. As I have said before in my editorials (not that anyone reads them I sometimes think) I had hope for around two hundred readers by the end of our first year. Well as you know it topped the 1000 mark. And its still continued to climb so that the number now stands at over 1600, making us the BIGGEST Spectrum user group in the WORLD.

The first 16 page issue was written entirely by yours truly and photo-copied, then hand collated and stapled. Now each issue is full of articles, mostly written by FORMAT readers, has a page count far larger than I ever thought possible. Since May 1988 FORMAT has been very professionaly printed and finished by D.S.Litho here in sunny Gloucester. Dave Smith (the D in D.S.) has often had to work late into the night to ensure FORMAT is published on time. My thanks to Dave (and to his wonderful wife and kids who struggle to my door each month with ever heaver boxes of FORMATS) for all the hard work.

As I said above, FORMAT is written for its readers and by its readers. I can't talk about authors without thanking a few by name. First my old mate John Wase. John is a professor of Bio-Chemistry but would much rather be closeted away in his spare bedroom working with his collection of Spectrums. He started writing for the old ZX Computing and did a lot to encourage the formation of INDUG and, of course, has contributed much to

FORMAT over the last two years.

Next comes Nev Young without whom several very important articles would never have been written. Nev has been in professional computing for longer than he likes to say but, with lots of encouragement from me he has become a regular contributor and has even been mad enough (sorry that should be KIND enough) to take on the roll of Agony Uncle with the Help Page each month.

Dick Guy is doing sterling work on the hardware front; an area I know little about. By reader demand FORMAT will now be carrying regular hardware articles including many construction projects and Dick will be my chief advisor in this area. So get your soldering irons heated up and keep reading FORMAT.

Ray Elder (school teacher and former editor of ZX Computing) has done much to foster understanding of computer music and MIDI interfacing. I hope to have more articles from him soon. Clyde Bish (another school teacher and former writter for ZX-C) has proved a firm favourite with readers of his superb series on Basic. Paul Rigby has produce an excellent column for Adventurers. Paul is a journalist in the aviation field but, since starting the Adventure Corner in FORMAT he has gone on to write for several other magazines. And I must not forget Ken Elston (who describes himself as a Jack of all trades - Master of propping up a bar) his INDEXER article this month saves me the very hard task of producing an index for FORMAT.

Last, but by no means least, Carol Brooksbank, one of my most prolific writters. I have more articles from Carol awaiting publication than any other author. Her SMALL IS BEAUTIFUL (Vol 2 No 5) was one of the most popular articles with readers this year. Its nice to see someone who really uses a Spectrum and can write so well about it.

Too all these writers, and to the many I have not had space to mention, I extend my personal thanks and the thanks of all FORMAT readers. Keep up the good work and may your keyboard never mistype.

Two years is a long time in computing, its also a long time in publishing. FORMAT has a secure future because it changes with the times. First as a DISCiPLE mag, then the introduction of the PLUS D. Now it covers the whole Spectrum field. And soon SAM will get a section of its own.

So what of the future? FORMAT enters its third year with the largest issue ever. A new logo (thanks to Dave Hood of Better Bytes) and a slightly new look. By the way, don't worry, I wont be writing such a big editorial every month. We will continue to grow, with the help of our readers. As I said in the first part of this editorial, if our readership grows so will FORMAT. But I always need more contributions. More programs (small and large), more Hints and Tips, more articles. If FORMAT is not printing articles on your pet subject then let me know, if I can find someone who knows about the subject then maybe an article will be forthcoming. I can only print what comes in but, if I know what readers want, I can at least try to find a writer to meet the demand.

Lets end on a high note with some idea of the articles already underway for volumn three.

128K Sound on the 48K; Faster than Basic; Spectrum Desk Top Publishing; Spectrum and the Radio Amateur

This is only a small list, a sample of the wide range of articles to come so keep your eyes open. FORMAT will continue to get bigger and bigger. Also, by popular demand as they say, our software service will be back next month with several new conversion tapes.

Thank you for reading and if you've still got that champagne in you glass, Good Health and here's to the next two years.

Bob Brenchley. Editor.

SHORT • SPOT

By: John Wase.

As an opener, here's a couple of brief tips. Harold Burton of Edinburgh (our faithful and prolific correspondent) has (bless him) come up with the goods again. His program sets up a printer through the DISCiPLE or PLUS D port for listings, allowing for the width of graphic characters by setting a left margin and limiting the line to 64 characters. Thus any graphics character merely causes the line to protrude a bit, instead of overflowing, as an 80 column line would. In addition, it also provides for a perforation skip: if the fanfold paper is not 11" long, you will have to amend the CHR$ 11 in the listing.

```
10 POKE @5,64: POKE @6,1: LPRINT CHR
   $ 27;CHR$ 64;CHR$ 27;CHR$ 108;CHR
   $ 8;CHR$ 27;CHR$ 67;CHR$ 0;CHR$ 1
   1;CHR$ 27;CHR$ 78;CHR$ 6;: POKE @
   6,0: RANDOMIZE USR 0
```

Stefan Limroth of Hamburg, Germany, adds a hint to the article in Vol 2 No 7 (p12), on the 128K version of "ArtistII" which prints with "The Writer", and in which were given some useful Pokes to inactivate the printer reset in the "Pagemaker" routine. Stefan mentions that in the 48k version of "Pagemaker", you need to change the "scrdump" file by Poking 64005 to 64010 with zero, and resaving the code (start=64000, length=107).

Now for another Really Useful Utility for 128K owners. Daniel Neidle of Watford, found that whilst the affluent with two disc drives can backup discs with "FORMAT d1 TO d2", mere penniless mortals are not only reduced to "SAVE d1"*" TO d1", (which takes ages), but have no way at all of copying microdrive type files: thus to make backups of, for instance, his 30 assembler files from "DevPac" proved impossible. Daniel"s utility asks for the number of Kilobytes to be copied, and then copies from disc to RAMdisc and back to disc again in 90K chunks (which is why it won't work on a 48K machine). Saves a lot of hassle, and will copy everything, including snapshots.

```
10 REM**** Disc Backup Routine ****
20 REM**** Daniel Neidle 1989 *****
30 REM Spectrum 128 or +2 only
40 REM Border turns red while Ramdis
   c "on"
50 REM
60 CLS#: BORDER 0: PAPER 0: INK 7: C
   LEAR 29997
70 PRINT AT 1,6; INK 6; BRIGHT 1;"Di
   sc backup routine"; OVER 1; AT 1,
   6;"                    "
80 INPUT "K-Bytes to copy? "; LINE a
   $ : IF a$="" THEN LET nt=208: GOT
   O 110
90 LET a= (VAL a$)+20: LET a=INT (a/
   5): LET nt=a: IF a>80 THEN LET a=
   a-80: LET nt=128+a
100 LET nt=nt+3: IF nt>208 THEN LET n
    t=208
110 PRINT #0; BRIGHT 1;" (c)DM Neidle
     June 1989"
120 LET t1=0: LET s1=1
130 LET r=0: LET t=t1: LET s=s1
140 PRINT AT 21,0; PAPER 6; INK 1; BR
    IGHT 1; FLASH 1;"Insert SOURCE di
    sc & press a key"
150 BEEP 1,1: PAUSE 0: PRINT AT 21,0;
    "{32 spaces}"
160 LET m=30000: BORDER 0
170 POKE 29998,t: POKE 29999,s
180 LOAD @1,t,s,m: LET m=m+512: IF m<
    6 5000 THEN LET s=s+1: LET s=s-(1
    0 AND s=11): LET t=t+(1 AND s=1):
     LET t=t+(48 AND t=80) GOTO 130+(
    10 AND t>nt)
190 BORDER 2: IF r=0 THEN SAVE !"1"CO
    DE 29998,35537: LET r=1: GOTO 160
200 IF r=1 THEN SAVE !"2"CODE 29998,3
    5537: LET r=1: GOTO 160
210 BORDER 0
220 PRINT AT 21,0; PAPER 6; INK 1; BR
    IGHT 1; FLASH 1; "Insert TARGET d
    isc & press a key"
```

```
230 BEEP 1,1: PAUSE 0: PRINT AT 21,0;
    "{32 spaces}": LET r=0
240 BORDER 0: LET i=PEEK 29999: LET m
    = 30000
250 SAVE @1,i,j,m: LET m=m+512: IF m<
    6 5000 THEN LET j=j+1: LET j=j-(1
    0 A ND j=11): LET i=i+(1 AND j=1)
    : LET i=i+(48 AND i=80): GOTO 200
    +(10 AND i>nt)
260 BORDER 2: IF r=0 THEN LET i1=i: L
    ET j1=j: LET r=1: LOAD !"1"CODE :
     ERASE !"1": GOTO 240
270 IF r=1 THEN LET r=2: LOAD !"2"COD
    E : ERASE !"2": GOTO 240
280 BORDER 0: LET t1=i1: LET s1=j1: L
    ET s1=s1+1: IF s1=11 THEN LET s1=
     1: LET t1=t1+1: IF t1=80 THEN LE
    T t1=128
290 IF t1>nt THEN NEW
300 GOTO 130
```

Nigel French of Spalding, noticed Robin Hughes' program for multiple disc formats, and was interested because he has written something rather similar, but which also numbers each disc, saving the last number to the master disc so that next time it carries on and numbers the next disc in order. Here is Nigel's program.

```
  1 REM * PLUS D DISC Numberer *
  2 REM *  By N.V.French 1989. *
  3 REM >>DO NOT RUN!<< Last Disc num
    ber will be ERASED! from memory.
    USE GOTO 80
 60 POKE 23658,8: REM CAPS LOCK
 70 LET LAST=0
 80 CLS #
 90 PRINT BRIGHT 1;AT 6,9;"DISC NUMBE
    RER";AT 8,8;"By Nigel French";AT
    10,13;"1989"
100 INPUT "PRESS ANY KEY TO START.";
    LINE A$
110 CLS: PRINT AT 0,0;"Last DISC numb
    er was ";LAST
120 INPUT "O.K. (y/n)? "; LINE Z$
130 IF Z$="Y" THEN LET NUM=LAST+1: GO
    TO 150
140 INPUT "ENTER NEW start number for
     DISC ";NUM
150 LET FLAG=0
160 PRINT "DISC number:-";NUM
170 INPUT "FORMAT (y/n)? "; LINE A$
180 IF A$="Y" THEN PRINT "FORMATING D
    ISC number:-";NUM: FORMAT d1: LET
     FLAG=1
190 PRINT "Saving SYSTEM FILE"
200 IF FLAG=1 THEN SAVE d1"+SYS 2 #"
    +STR$ NUMCODE 8192,6656: GOTO 220
210 ERASE d1"+SYS 2*": SAVE d1"+SYS 2
     #"+STR$ NUMCODE 8192,6656
220 CAT 1
230 INPUT "O.K. (y/n)? "; LINE A$
240 IF A$="Y" THEN LET LAST=NUM: LET
    NUM=NUM+1
250 INPUT "Quit (y/n)? "; LINE A$
260 IF A$="N" THEN CLS : GOTO 160
270 CLS : INPUT "Do you wish to SAVE
    the current DISC NUMBER (y/n)? ";
     LINE A$
280 IF A$="N" THEN STOP
290 CLS
300 PRINT AT 20,0;"Insert the DISC wi
    th the ORIGINAL DISC num program
    on it.": INPUT "Then press ENTER.
     "; LINE Z$
310 SAVE d1"DISC num" LINE 110: STOP
330 REM 1st SAVE
340 SAVE d1"DISC num" LINE 10
```

You know, "FORMAT" gets a pretty incredible circulation. Roy Burford of Norton, who has a Tandy TRS80-II, was idly reading a friend's "FORMAT" (Vol 2 No 12)) and spotted the item by Nigel Baumann on INKEY$: you remember; it let you put in more than one character. Here's his routine (for the TRS80) which gives the same result.

```
 0 'Inkey routine for more than one c
   haracter.
50 PRINT "Type in the number required
    from 1 to 22 followed by a single
    space:"
55 C$="":'Null string. LET optional o
   n TRS80-II Micro-Computer
60 M$=INKEY$
70 IFM$=""THEN60:'Null string loop
80 IFM$=" "THEN120:'Conditional branc
   h -out
90 C$=C$+M$:'Concatenate(add) strings
100 IFINT(VAL(C$))<>VAL(C$)THEN40: 'Ch
    eck whole number
105 IFVAL(C$)<1ORVAL(C$)>22THEN40: 'Ch
    eck within range
110 GOTO60:'Get next digit
120 PRINT:PRINT C$
130 END
```

That's all for now and I've finished in time, so I can go on my hols with a clear conscience. (Well, fairly)! Please keep the contributions coming in: see you next month!

ADVENTURE CORNER

By: Paul Rigby.

A letter! A letter! (sounds of hysterical laughter) Ahem. Yes, folks I have received a letter. I knew you could do it, or rather Phil Glover of Birmingham has. Thanks Phil! Reading through Phil's letter I noticed that he comments about how he actually plays adventures. He says that, "I tend to make fair progress, then swap to another game for a change. I tend to read books the same way, slowly chugging through half a dozen books at various times."

Which is interesting as I play adventures (and read books for that matter) in exactly the opposite way. I play one adventure at a time telling myself that I cannot begin another until I finish the present one. I assume that the whole question of Adventure Psychology comes into focus here. A subject I mentioned in the first Adventure Corner back in Vol 2 No.2. The subject is an interesting one. Is it due to impatience that some adventurers leap from one unfinished adventure to another? Is it because a large percentage of adventures are just too difficult? Do some adventurers find adventures easier to complete than others? After all playing adventures requires more thought, lateral, logical and, unfortunately, illogical thinking than your average arcade game.

You've done it before. After breezing through a few initial puzzles, merrily clocking up the score, you suddenly hit rock. A puzzle no, then again, an absolute brick wall appears to bar your way. You are stuck. You sit and stare at the screen with a rapidly depressing look covering your face like so many black clouds covering the sky. What do you do? Some players try, for hours, weeks and months, to battle through the problem. Others give up after two minutes. When you do hit a problem, and I am talking to those people who immediately reach for the adventure's solution when they discover a problem. Please give the puzzle a chance. That is, try to solve it on your own. Try developing a set system which you employ once you arrive at a particularly tough puzzle. Go through each stage of your system, methodically working your way through your guide. Then, if you still do not have any success, reach for the telephone, get in touch with a friend or look through the glossies for a hint to get you through the puzzle. There will, probably, be many adventurers who disagree with me on this point, but I cannot see how anyone can be stuck on the same puzzle, in the same adventure, for months on end and still be enjoying it as a piece of entertainment. At that stage of the proceedings I would have lost all interest, would have forgotten what was going on anyway and how I reached the present sticky position and would have lost the essential atmosphere that any story brings with it.

So what sort of system should you develop? Well, the final system should be one of your own choosing as everyone has their own way of playing adventures. However, maybe I could make one or two suggestions to help you on your way.

When I reach the "brick wall" I tend to try different methods of approaching the problem. Let us, for example, have the scenario where we are trying to pass a guard to venture further into the game. Maybe to enter a castle. Let us assume that the only

way into the castle is by using the entrance that the guard is standing by. Maybe he requires an object from your inventory before he will let you pass. So you give him the jewel instead of the jam roly-poly you initially tried. But no he will not budge. So you try hitting, killing, stabbing and punching him (I do alot of this when I am really stuck - I think its called desperation) but no - no success. You could track backwards a few locations and thoroughly examine the location descriptions seeing if you missed anything. Maybe you forgot about one of the exits. This has happened to me once or twice. After being stuck in an adventure called "Appleton" I called a friend on the telephone in desperation only to be asked if I had tried going North from one of the locations? No, I said meekly, I hadn't. I had succeeded in creating my own puzzle in addition to the puzzles presented by the game!

There are some very sneaky adventures which, to pass a particular point, will allow you to forfeit any one of two or three objects from your inventory. This means that you could possibly give away the wrong object which you will need later on (to give to the guard, for example). Be wary of this situation. You would need to take regular saves of your adventure to allow you to quickly check if such a trap has been sprung.

Another avenue you should try, although no good game will have you in this position, is to try the illogical commands, something I touched upon above. A good source of illogicality is Scott Adams. A man who always said that his adventures were logical. I totally disagree with him. In fact I think Scott Adams has probably done the most harm to the adventure cause than any other single individual. My evidence, your Honour, is, amongst others, the series of games which depicted the Marvel Superheroes. I hated them. Half decent graphics, but terrible games. I, personally, am relieved that he has taken leave of adventure authoring to concentrate upon other matters. The sad thing is

that many potential adventurers will have found their first adventure a Scott Adams production and would have been put off for life.

A more recent case of illogicality is Eric Stewart's home-grown adventure "The Legend Of Craldon's Creek" which was published around February/March 1988. One location presented you with a magical force which prevented you from going in one direction. All (!) you had to do was to drop a rotten apple in the "magical force" location and the latter would disappear! No logical reason was given for the occurrence but that was the answer to the problem. Adventures comprise of good and bad examples. It is, without suitable advice, pure luck as to what the beginner will stumble across when they take up the hobby. But, if there is anyone out there who has been bitten by a bad adventure then don't give up! There are an awful lot of good examples out there which give genuine enjoyment which brings you back for more. Now there's a thought. How about you sending in your top five or top ten adventures to act as a guide to the beginner. I am sure many adventurers starting out on the trail would appreciate it. I'll collate my own list too for a future issue.

Back to that guard we were trying to pass. Don't forget passwords, long words which mean no apparant sense, strange phrases and other, similar happenings. Sometimes these will be spoken early on in the game. Write them down! You just never know when it will come in handy. Coded messages are another source. If a character says, or you see written on a tree, a string of meaningless letters, again, write them down. Chances are that it will be a code. I will go into greater depth with coded messages in a future Corner but for now - eb erawa! (sorry - be aware!)

Whatever system you use give the puzzle a chance before you give up. The author, if the game and the particular puzzle is a good one, will have put a great deal of thought into it. The puzzle may be a work of great imagination so it is a shame not to try to overcome it. I know that I have experienced great satisfaction whenever I have solved a good puzzle. It is not a case of defeating the programmer, that should not figure at all, it should be more of a matter of over-coming the game environment, or dastardly character or somesuch. Many friends and colleagues have then been bored to tears by the fifth account of how I overcame the Troll by the bridge, "There was I - a flint, a lamp and alot of hope..."

In the May '89 (Vol 2 no.9) issue I mentioned that it would be nice to see magazines and/or fanzines published by software houses. I also mentioned that it would be interesting to see the smaller software house contribute towards this aim. When I wrote those words I originally thought about the individual software houses publishing some information. However, in his letter, Phil Glover suggests that matters could be taken further, "I'd like to see an association formed by and for the benefit of small adventure publishers. Advertising must be expensive, but if they clubbed together to produce, say, a quarterly news sheet with names and addresses and adverts of their adventures it may prove economic, and provide plenty of information for customers."

That's a good idea Phil. Taking the idea further still - if some sort of unofficial co-operative scheme where available for adventure authors and small software companies I am sure that the collective effort would help to reach more people. I also believe that a co-operative could be the basis for a group to help and encourage those people who have the enthusiasm, a slice of talent but do not have a clue how to survive in the big bad world of marketing, for example. Many old hands have learned from mistakes which have, luckily for them, not done too much damage in the long term. I do know that similar misfortunes (mainly of the financial nature) discourage many others who close shop. Maybe those same "old hands" could pass on their experience via an association

publication. Possibly, an advice service could be developed (a problem shared...and all that) collective funds may be able to purchase expensive equipment to improve the packaging, for example, of adventures. I do believe, though that significant and recognisable software houses (such as Eighth Day) and figures (such as Tom Frost) should be in at the beginning to give such an organisation some credibility. Do any readers out there have any thoughts on the matter?

Well, I've rambled on for long enough, I'll return to Phil Glover's letter (Format readers are full of good ideas - why don't a few more of you drop me a line. It can be about anything remotely connected to adventures. It doesn't have to bear any relevance to what I've talked about this month, or any month for that matter) so goodbye until next month and don't forget - keep it Cornered!

INDEXER

By: Ken Elston.

Ever wanted an index to past issues of FORMAT? Well here is the answer to all your prayers (NO... not a printed index, that would be too easy) a program so that you can build your own index for FORMAT or any other magazine for that matter.

The program uses the array (A$) to hold details of the articles you want to index. M$ holds the names of up to twenty magazines while C$ holds the names of up to 20 categories you can search for. The details you can store on each article are Category, Magazine, Month & Year, Page, and Title. Some fields are stored as 1 byte numeric, this means that you can get any number in the range 0-255 into one byte, which keeps the memory requirement down. Let's say you want to store 164. To encode the number use:- **LET A$(n)=CHR$(164)** and to decode use:- **LET X=CODE(A$(n))** Nothing could be easier could it?.

Start by RUN 1000. After asking you for the number of magazines, and categories you want, a calculation is made of the free space available for your data (line 1220). If you change the DIMmension of A$ you will need to alter the number 22 in that line. The rest of the program should be fairly easy to follow. The program will work on any Spectrum and the SAVE/VERIFY syntax could be changed to tape, microdrive or other disc systems.

```
  1 REM INDEXER. (c)1989 FORMAT.
  2 REM A$(1)=CODE CATEGORY
  3 REM A$(2)=CODE MAGAZINE
  4 REM A$(3 TO 6)=MMYY
  5 REM A$(7)=CODE PAGE NUMBER
  6 REM A$(8 TO )=TITLE/INFO
 10 POKE 23658,8: GOSUB 3000: PRINT '
    "OPTIONS AVAILABLE:-"'''"1 - Add N
    ew Entry"'''"2 - Amend Entry"'''"3
    - View By Category"'''"4 - View By
     Magazine"'''"5 - Save File"'''"6 -
     Start New File"
 20 LET I$=INKEY$: IF I$="" THEN GOTO
     20
 30 IF I$<"1" OR I$>"6" THEN GOTO 20
 40 GOSUB 100*VAL I$: GOTO 10
100 REM ADD RECORD
110 IF NO>MAX THEN PRINT #0; FLASH 1;
    "NO MORE ROOM": PAUSE 1: PAUSE 0:
     RETURN
120 LET NO=NO+1: LET N=NO: GOSUB 2100
    : GOSUB 2500: GOSUB 2200: GOSUB 2
    300: GOSUB 2400
130 GOSUB 2000
140 INPUT "IS THIS OK?  Y or N";I$: I
    F I$="Y" OR I$="y" THEN RETURN
150 LET NO=NO-1: GOTO 100
200 REM AMEND ENTRY
210 GOSUB 3000: FOR I=1 TO NO: PRINT
    I;TAB 4; INVERSE 1;A$(I,8 TO ); I
    NVERSE 0;C$(CODE A$(I,1)): NEXT I
220 INPUT "ENTRY NUMBER OR 0 TO REPEA
    T ";N: IF N=0 THEN CLS: GOTO 210
230 IF N>NO THEN GOTO 220
240 GOSUB 2000: PRINT '''"PRESS NUMBER
     TO AMEND"'' INVERSE 1;"0 TO  RET
    URN"
250 LET B$=INKEY$: IF B$="0" THEN RET
    URN
260 IF B$<"1" OR B$>"5" THEN GOTO 250
270 CLS: GOSUB 2000+VAL B$*100
280 GOTO 240
300 REM VIEW BY CAT
310 GOSUB 3000: FOR I=1 TO C: PRINT I
    ;TAB 6;C$(I): NEXT I
320 INPUT AT 0,0; INVERSE 1;"ENTER NU
    MBER OF CATEGORY        0 TO RETU
    RN";N: IF NOT N THEN RETURN
330 IF N>C THEN GOTO 320
340 CLS: PRINT  INK 0; PAPER 7;"CATAG
    ORY - ";C$(N)': FOR I=1 TO NO: IF
     CODE A$(I,1)=N THEN PRINT I;TAB
    5;A$(I,8 TO )
350 NEXT I: INPUT "ENTER NUMBER FOR I
    NFORMATION    0 TO  RETURN";N: IF
     NOT N THEN RETURN
360 IF N>NO THEN GOTO 350
```

```
370 GOSUB 2000: PRINT #0;"ANY KEY": P
    AUSE 1: PAUSE 0: RETURN
400 REM VIEW BY MAG
410 GOSUB 3000: FOR I=1 TO MAG: PRINT
     I;TAB 6;M$(I): NEXT I
420 INPUT AT 0,0; INVERSE 1;"ENTER NU
    MBER TO VIEW ENTRIES     0 TO RETU
    RN";N: IF NOT N THEN RETURN
430 IF N>MAG THEN GOTO 420
440 GOSUB 3000: PRINT M$(N)'': FOR I=
    1 TO NO: IF CODE A$(I,2)=N THEN P
    RINT I;TAB 5; INVERSE 1;A$(I,8 TO
     )
450 NEXT I: INPUT "ENTER NUMBER FOR I
    NFORMATION     0 TO RETURN";N: IF
    NOT N THEN RETURN
460 IF N>NO THEN GOTO 450
470 GOSUB 2000: PRINT #0;"PRESS ANY K
    EY": PAUSE 1: PAUSE 0: RETURN
500 REM  SAVE PROG & DATA
510 SAVE d*"MAG.DAT" LINE 10
520 VERIFY d*"MAG.DAT": RETURN
600 REM NEW FILE
610 GOSUB 3000: PRINT "This option wi
    ll overwrite your current file."
620 INPUT  FLASH 1;"ARE YOU SURE"; FL
    ASH 0;" Y or N ";I$: IF I$="Y" TH
    EN RUN 1000
630 RETURN
1000 REM INITIALISATION
1010 GOSUB 3000: INPUT "HOW MANY MAGAZ
     INES? (1-20) ";MAG
1020 IF MAG<1 OR MAG>20 THEN GOTO 1010
1030 PRINT "NUMBER OF MAGS = ";MAG''
1040 DIM M$(MAG,15)
1050 FOR I=1 TO MAG
1060 INPUT "NAME OF MAGAZINE ";STR$ I;
     "? (15 letters)"'M$(I)
1070 PRINT M$(I);" ";: NEXT I: PRINT
1100 INPUT "IS THIS OK?  Y or N";I$: I
     F I$="Y" OR I$="y" THEN GOTO 1130
1110 IF I$="N" OR I$="n" THEN RUN 1000
1120 GOTO 1100
1130 GOSUB 3000: INPUT "NUMBER OF CATE
     GORIES? (1-20) ";C: IF C<1 OR C>2
     0 THEN GOTO 1130
1140 DIM C$(C,15)
1150 PRINT "NUMBER OF CATEGORIES = ";C
     ''
1160 FOR I=1 TO C
1170 INPUT "CATEGORY-";STR$ I;" DESCRI
     PTION"'C$(I)
1180 PRINT C$(I);" ";: NEXT I: PRINT
1200 INPUT "ARE THESE OK?  Y or N";I$:
      IF I$="Y" OR I$="y" THEN GOTO 12
     20
1210 IF I$="N" OR I$="n" THEN GOTO 113
     0
1220 LET I=INT (((64536-USR 7962)-8)/2
     2)
1230 GOSUB 3000: INPUT "NUMBER OF ENTR
     IES? (MAX=";STR$ I;") ";MAX
1240 IF MAX>I THEN GOTO 1230
1250 DIM A$(MAX,22): LET NO=0: GOTO 10
2000 REM  PRINT RECORD
2010 GOSUB 3000: LET T=VAL "13": PRINT
      '"1-CATEGORY";TAB T;C$(CODE A$(N
     ,1))'"2-MAGAZINE";TAB T;M$(CODE A
     $(N,2))
2020 PRINT "3-MONTH/YEAR";TAB T;A$(N,3
      TO 4);"/";A$(N,5 TO 6)'"4-PAGE";
     TAB T;CODE A$(N,7)
2030 PRINT "5-TITLE";TAB T;A$(N,8 TO )
2040 RETURN
2100 REM SELECT CATEGORY
2110 GOSUB 3000: FOR I=1 TO C: PRINT I
     ;TAB 5;C$(I): NEXT I
2120 INPUT "WHICH CATEGORY? ";A: IF A>
     C THEN GOTO 2120
2130 LET A$(N,1)=CHR$ A: RETURN
2200 REM  INPUT MAG
2210 GOSUB 3000: FOR I=1 TO MAG: PRINT
      I;TAB 6;M$(I): NEXT I
2220 INPUT "WHICH MAG?";A: IF A<1 OR A
     >MAG THEN GOTO 2220
2230 LET A$(N,2)=CHR$ A: RETURN
2300 REM  INPUT MONTH & YEAR
2310 INPUT "MONTH NUMBER (ie 3 or 11)"
     ;A: IF A<1 OR A>12 THEN GOTO 2300
2320 LET i$="00"+STR$ A: LET I$=I$(LEN
      I$-1 TO )
2330 LET A$(N,3 TO 4)=I$
2340 INPUT "ENTER YEAR (i.e. 89)";A
2350 LET i$="00"+STR$ A: LET I$=I$(LEN
      I$-1 TO )
2360 LET A$(N,5 TO 6)=I$
2370 RETURN
2400 REM  INPUT PAGE
2410 INPUT "PAGE NUMBER (0-255)";A: IF
      A<0 OR A>255 THEN GOTO 2400
2420 LET A$(N,7)=CHR$ A
2430 RETURN
2500 REM  INPUT TITLE
2510 GOSUB 3000: INPUT  INVERSE 1;"ART
     ICLE TITLE?"; INVERSE 0'A$(N,8 TO
      )
2520 RETURN
3000 REM PRINT HEADING
3010 PAPER 1: INK 7: BORDER 1: CLS
3020 PRINT  INK 0; PAPER 6; BRIGHT 1;"
     MAGAZINE INDEXER - (C)1989 FORMAT
     ."'':REM (C) is copyright sign (E
     xtended Mode + Sym/Shift P.)
3030 RETURN
9999 SAVE d1"MAG-INDEX" LINE 1000
```

THE SAM SPOT

By: Bob Brenchley.

Still no firm release date for the SAM Coupe computer, but things can't be far off now. Bruce's Mega Chip (BMC for short) has passed all tests with flying colours and a small number of hand-built machines are now being used to develop the final version of the ROM and to work on the graphics package that will be bundled with the machine (see last month).

Don't believe everything you see in other magazines, nobody has been allowed to see the finished machine yet. The case is still undergoing final design changes. Bruce is working on the final board layout and the keyboard is still a couple of weeks away from existing in its real form. However, that is not to say that SAM dosn't exist, what I'm trying to say is there is still a lot of things awaiting delivery from outside contractors befor anyone can say "I've seen a finished SAM". So take what some magazines say with a pinch of salt.

So whats new? Well Alan Miles and Bruce Gordon were invited to give a presentation to the European Leisure Software Publishers Association (ELSPA) last month. Representatives of around twenty-five software companies were present and there was great interest in SAM. The list of software companies interested in producing SAM dedicated software (or conversions from other formats) grows longer each week. After so long without a new 'Mass Market' machine the UK software industry looks forward to SAM with much anticipation.

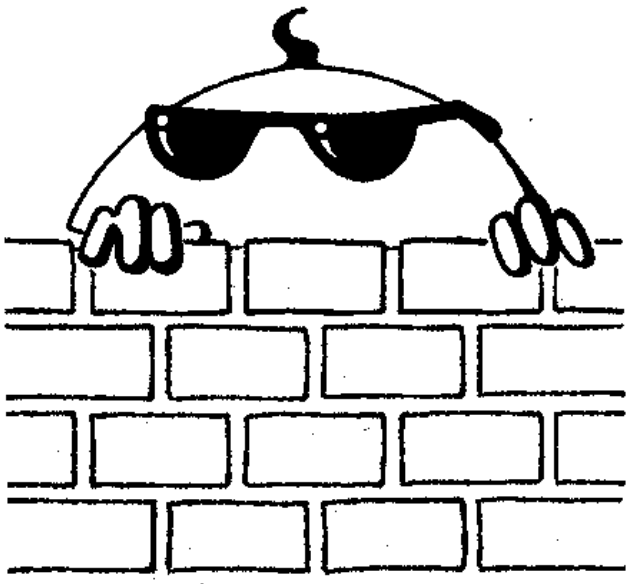
MGT have also announced several steps to ensure software companies can get to grips with SAM as quickly and easily as possible. Last month we told you all about Bo Jangeborg's Art package, well that will be very important to software writers. But also of importance is the music side, with a sound chip as advanced as the Phillips SAA1099, no SAM game will be complete without fantastic stereo sound. So MGT have commissioned David Whittaker to produce music and sound effects development software for use by software houses.

David, pictured above, has been responsible for the sound in hundreds of games including the likes of Platoon, Tetris and Licence to Kill. His SAM software will include machine code 'drives' to turn a stream of data into the controls for the six channel sound chip. He will also write an emulator for the old GI sound chip used in many existing computers.

Many software companies use a development system called PDS. This

allows them to write software, on old fashioned machines called IBM PCs, using assemblers and debugging tools aimed at there target machine. The system relies on an interface and support software on the target computer and this is already being written for SAM. Altogether its going to be pretty easy for software houses to get started with SAM and few other hardware manufactures have done as much as MGT to help encourage software for a new machine.

Alan Miles has also asked me to pass on the news that the long-awaited brochure for SAM is now nearing completion. Anyone who has registered an interest in SAM will be sent one, about one month before the computer is due to be launched. It now looks as if MGT will be using distributors, to get SAM into a network of dealers, as their main method of selling. But people who are on MGT's list will be guaranteed delivery direct, before shipments are made to dealers, so make sure you register now.



WHO'S THIS LOOKING
OVER THE WALL?

Read Next Months FORMAT
If You Want To Find Out..

THE SECRETS OF WORD MANAGER

SPECTRUM MACHINE CODE MADE EASY

Part 5. **By: Francis Miles.**

LOOPS. - Part 1.

"Word Manager" has a routine for jumping the cursor forwardby a whole screen. It uses a subroutine PLUSL which moves the cursor on one print line; it's just a question of calling PLUSL the right number of times. For a 64-character line this is 24 times, but if the printline is more than 64 characters and so takes up two screen lines it will be 12 times; 8 times if the printline takes up 3 screen lines and 6 times if it takes 4 screen lines. At the start of this excerpt A has been loaded with a byte from the "Word Manager" system variable P.S which records "lines per line", 1, 2, 3 or 4.

```
6380 ;Load B with number of lines.
6390         LD B,24
6400         DEC A
6410         JR Z,FD.LP
6420         LD B,12
6430         DEC A
6440         JR Z,FD.LP
6450         LD B,8
6460         DEC A
6470         JR Z,FD.LP
6480         LD B,6
```

[One of the DEC A's in lines 6400, 6430 and 6460 will have reduced A to zero, so the program jumps to the loop with the right value in B.]

```
6490 ;Move cursor on B lines.
6500 FD.LP   PUSH BC
6510         CALL PLUSL
6520         POP BC
6530         DJNZ FD.LP
```

[PLUSL messes up all the registers, so the counter must be saved on the stack anyway while it is called.]

Notice that the DJNZ command is very much more flexible if its counter is put on the stack and only POPed at the last minute. Consider the following, which is what happens in "Word Manager" if you input so much text that you run out of memory to store it. The program clears the screen and prints the message:-

Not enough memory
Last command ignored

Then it does five short BEEPS each separated by a short pause, programmed as follows:-

```
1310         LD B,5
1320 ;Outer loop - beep and pause.
1330 OF.LP1  PUSH BC
1340         CALL TONE;standard BEEP
1350         LD B,10
1360 ;Inner loop - pause.
1370 OF.LP2  HALT
1380         DJNZ OF.LP2
```

[HALT is a one-fiftieth second delay, so ten passes through this loop is a little more than a fifth of a second.]

```
1390         POP BC
1400         DJNZ OF.LP1
```

Here we have nested loops, but the counters don't interfere with each other because the counter for the outer loops is stacked while the inner loop is run. However, the longest and fanciest loops in "Word Manager" don't actually use DJNZ at all; one can write perfectly good loops with JR or JP instructions, using any register you like as a counter. The "Word Manager" subroutine PAGE is called every time a key is pressed when text is on screen, and it "paints" 24 lines of text on the screen. I will show excerpts from PAGE next month, but some preliminary comments are

necessary. The problems with PAGE were

1. None of the ROM routines are much help, as they're only geared to print 32 characters per line.

2. The routine has to be lightning fast, because it has to be able to keep up with a fast touch-typist.

3. The Spectrum screen buffer is arranged in a peculiar way. Each character is made up of eight bytes one above the other like this:-

```
        ........
      A ..XXXX..
        .X....X.
        .X....X.
        .XXXXXX.
        .X....X.
        .X....X.
        ........
```

On the screen the Xs show as "dot pixels", the points as "blank pixels".

The screen is divided into "thirds". The eight bytes of a character are not stored consecutively in the screen buffer; it stores first all the top bytes of the first third, then all the second-row bytes, then the third-row and so on; then all the top bytes of the second, third, and so on. Each third contains 8 lines of 20 hex characters (it is easier to describe the screen using hex numbers. Never mind their decimal equivalents for the moment, though you may remember that 20 hex is 32 decimal). This is 8 x 20 hex = 100 hex characters, and each has 8 bytes, so each third consists of 800 hex bytes altogether. The first third starts at 4000 hex, and the numbers run like this:-

```
First line, top row bytes:
     4000 hex, 4001 hex, ..., 401F hex
Second line, top row bytes:
     4020 hex, 4021 hex, ..., 403F hex
Third line, top row bytes:
     4040 hex, 4041 hex, ..., 405F hex
etc
Eighth line, top row bytes:
     40E0 hex, 40E1 hex, ..., 40FF hex

First line, 2nd row bytes:
     4100 hex, 4101 hex, ..., 411F hex
Second line, 2nd row bytes:
     4120 hex, 4121 hex, ..., 413F hex
Third line, 2nd row bytes:
     4140 hex, 4141 hex, ..., 415F hex
etc
Eighth line, 2nd row bytes:
     41E0 hex, 41E1 hex, ..., 41FF hex
etc
etc
First line, 8th row bytes:
     4700 hex, 4701 hex, ..., 471F hex
Second line, 8th row bytes:
     4720 hex, 4721 hex, ..., 473F hex
Third line, 8th row bytes:
     4740 hex, 4741 hex, ..., 475F hex
etc
Eighth line, 8th row bytes:
     47E0 hex, 47E1 hex, ..., 47FF hex
```

Then the second third starts at 4800 hex, exactly similar, and the last at 5000 hex (ending at 57FF hex).

It was fairly obvious that PAGE must contain at least a triply nested loop, one loop each for thirds, characters and bytes. Also it was clear that finding the character in the specially drawn "condensed" character sets (see below) is a relatively slow operation, so it would be better to paint all its bytes to the screen at once rather than look for it each time. This pretty well determined the order of nesting.

"Word Manager" contains two specially drawn character sets,compressing each letter into a rectangle three bits wide (leaving one bit blank for a space between characters). For example lower-case "s" in the "right font" is:-

```
        ........
      s ........
        ........
        ......XX
        .....X..
        ......XX
        .......X
        .....XX.
```

(The eight pixel bytes of the letter are stored consecutively in the character set.)

The same letter in the "left font" is just the same except that each byte

is rotated 4 places left (multiplied by 16). It may seem wasteful to repeat the character set in this way; but if (say) the left font was calculated from the right font, the calculation would have to be done for each of 7 bytes of 32 characters (half the total) on 24 lines: 7 x 32 x 24 = 5376 times, every time any key is pressed; there just isn't any time to spare for doing arithmetic with the byte once you've found it.

What the program does is load four pixels of the first character from the text into the high nibble of a byte, and four pixels of the second character into the low nibble of the same byte, then poke this double character byte into a position on the screen. Suppose the two letters to be put on the screen are "th". Then DE is pointed at "t" in the left font:-

```
                ........
          DE -> ........
t (left font) ..X.....
                .XXX....
                ..X.....
                ..X.....
                ...X....
                ........
```

and HL is pointed at "h" in the right font:-

```
                ........
          HL -> .....X..
h (right font) .....X..
                .....XXX
                .....X.X
                .....X.X
                .....X.X
                ........
```

Then the byte in DE and the byte in HL are combined into a pixel byte for a double letter, and poked to the screen as:-

```
                ........
     DE & HL -> .....X..
             th ..X..X..
                .XXX.XXX
                ..X..X.X
                ..X..X.X
                ...X.X.X
                ........
```

There is a slight shortage of registers. DE gets the address of the left font character form, and HL that of the right font form; and one more register (BC) is needed to calculate these addresses. When we've got the addresses we combine their pixel bytes in A and poke the double letter pixel into the screen position, which we keep on top of the stack. Where are we going to keep the text position? The answer adopted is to use the "alternate register set". The command EXX brings in an entirely new set of registers replacing BC, DE and HL (EXX again brings the old ones back, with their values unchanged), while leaving the stack and the AF register unchanged. A vital point is that EXX is very fast. I could have juggled with the stack to keep both the text position and the screen position on it; but a PUSH followed by a POP takes 21 cycles of chip time; EXX only takes 4.

EXX is not used very often, and is quite difficult to handle, but it neatly fills the bill in this subroutine.

More about PAGE next month.

Don't worry, I always use the wife when I forget my calculator.

SPECTRUM SPRITES

By: Carol Brooksbank.

There are a number of handy programs around which help to take the pain out of sprite design and animation. Each one has its good and bad points, so I hope this brief summary of some of them will help you to decide which is likely to suit you best. I have used the same headings for each program, to help you to compare them, and wherever possible there is a screen shot.

Where the sprite generator is part of a graphics package, I am only evaluating the sprite generator and not the program as a whole, and where the sprite generator is a separate program from the main one, the instructions about transfer to disc refer only to the sprite program.

LASER BASIC. - Ocean (Fig.1)

1) OVERVIEW:-
Program for sophisticated and complex sprite animation, using extended BASIC commands.

2) SPRITE GENERATOR SEPARATE? :- Yes.

3) TRANSFER TO DISC:- Easy.
Load generator from tape. Break into BASIC and MERGE the BASIC lines given in Program 1. (Purists can also change M/DRIVE in lines 3650 and 3750 to DISC). Enter GOTO 9998 and program will be saved to disc. Do not try to use the 'Format Cartridge' option. It will only give an error report, and anyway, with the disc's greater capacity, you are not limited to 5 sprite files as with Microdrive.

4) MAX SPRITE SIZE (character sqrs):-
15x15. Facility for larger sprites and storing them in memory, but they cannot be displayed on generator screen all at once.

5) MEMORY STORAGE CAPACITY:-
12K of memory reserved for sprite file. Maximum of 255 frames in a file, depending on sprite size. Continuous display of number of free bytes.

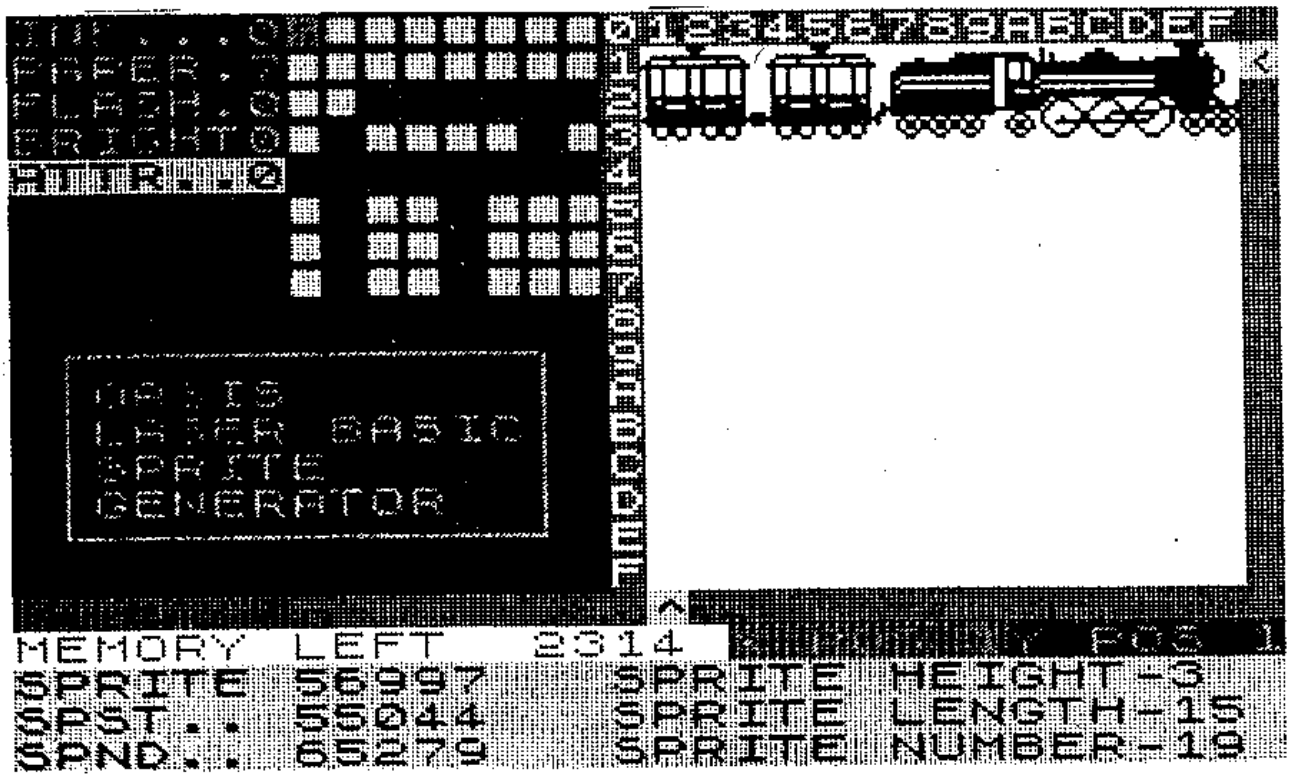


Fig 1

6) ANIMATION TEST?:-
Not from sprite generator. You need to save sprite file in two forms - one to load into main program for testing, the other in case you need to load back into the generator to modify any frames.

7) ADVANTAGES:-
Accepts numerical input in hex or decimal, as well as design by setting and resetting pixels on enlarged grid. Large sprite file capacity ideal for building a sprite library. Comes with 2 files of sprites, giving a library of 109 frames, some of which are related frames for animation.

8) DISADVANTAGES:-
Most of program in BASIC, so response to keyboard slow. Sprite is designed one character square at a time, and the square must then be dumped to the display screen, after moving the pointers to the required place. No animation test in generator. Need to save sprite file in two ways.

9) VERDICT:-
Laser Basic, with its companion Compiler, makes sophisticated sprite animation available to those who do not know machine code. The sort of movement you find in games like "Manic Miner" is easily achieved with this program, in surprisingly few lines of BASIC.

ART MASTER - Summit (Fig. 2)

1) OVERVIEW:-
Extended BASIC art package.

2) SPRITE GENERATOR SEPARATE?:- No.
Sprites designed as part of screen.

3) TRANSFER TO DISC:- Very simple.
Follow instructions in handbook for transfer to Microdrive. All SAVE/LOAD done from BASIC as direct commands, so no modifications required.

4) MAXIMUM SPRITE SIZE:- Full screen

5) MEMORY STORAGE CAPACITY:-
Extraordinary. When compiled, the code is highly compressed. The 14x12 sprite in Fig. 2 uses 119 bytes, because the number of commands required to draw it governs the memory needed, not its size. Saved as a string of bytes, the same sprite would need 1344 bytes.

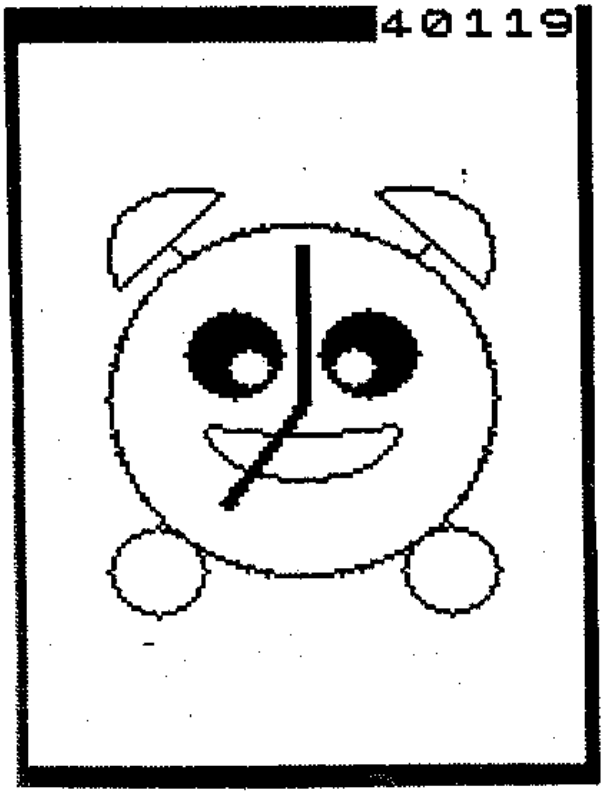


Fig 2

6) SPRITE ANIMATION TEST?:- Yes.
BASIC must be compiled (a program option) to a user-selected address, then its running can be tested, with a continuous display of the memory address being executed.

7) ADVANTAGES:-
No knowledge of machine code needed to produce animated program. Efficient compression means huge number of sprites and screens can be held in memory.

8) DISADVANTAGES:-
Sprite must be planned on graph paper, and co-ordinates etc. determined before programming. (Program 2 is the listing for the sprite in Fig. 2). Even when compiled, the program runs very slowly. Every line and fill added can be seen. Only very rudimentary movement is possible. As sprite is programmed in BASIC, there is no display of the sprite as you build it up. Program must be compiled and tested to check for drawing errors.

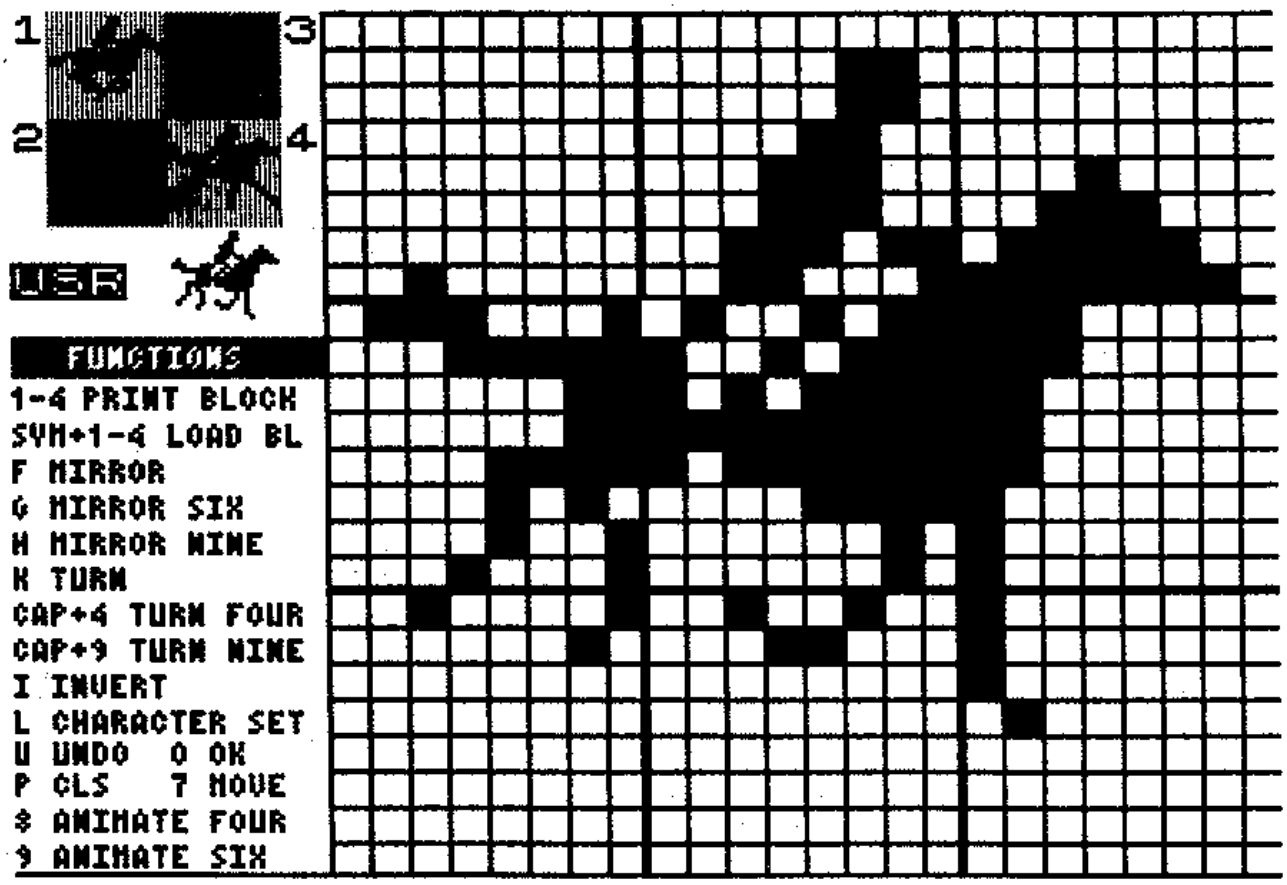


Fig 3

9) VERDICT:-
Not suitable for sophisticated work, but if storage of a large number of sprites and screens is more important than complex animation, this program can be useful.

THE ARTIST - Softek (Fig. 3)

1) OVERVIEW:-
Graphics package. Forerunner of THE ARTIST 2.

2) SPRITE GENERATOR SEPARATE?:- No.
A main menu option.

3) TRANSFER TO DISC:-
Reset Spectrum - if using 128K Spectrum, select 48K mode.
MERGE first program on tape.
Alter loading instruction in line 6 to: LOAD d1 "artscr" CODE 43008
Alter loading instruction in line 7 to: LOAD d1 "artbas"
SAVE d1 "artist" LINE 0
Enter CLEAR 43007
LOAD code block from tape
SAVE d1"artscr" CODE 43008,5200
Reset Spectrum - select 48K mode
MERGE next program from tape
Alter loading intruction in line 60 to: LOAD d1"artcod" CODE
SAVE d1"artbas" LINE 60
Enter CLEAR 49919
LOAD code block from tape
SAVE d1"artcod" CODE 49920,15615
To load the program from disc: LOAD d1 "artist"

NB. Don't try to use this program in 128K mode, or selecting the sprite generator will cause a crash.

4) MAX SPRITE SIZE (character sqrs):-
3x3.

5) MEMORY STORAGE CAPACITY:-
Limited. To store more than half-a-dozen frames, you must over-write main program typefaces or fill patterns.

6) SPRITE ANIMATION TEST?:- Yes.
Animate 4 or 6 frames at fixed speed.

7) ADVANTAGES:-
Full-size enlarged grid. Sprites can be grabbed from or put into screen in main program - useful for adding textures or building up large sprites. Four related frames can be displayed in addition to one being worked on.

8) DISADVANTAGES:-
Very limited memory storage. Limited sprite size Arrow keys are not used for cursor movement.

9) VERDICT:-
My favourite designer for small sprites. I like especially the full-size grid and the facility for having related frames on screen for comparison while you work.

THE ARTIST 2 - Softek (Fig.4)

1) OVERVIEW:-
Sophisticated art package.

2) SPRITE GENERATOR SEPARATE?:- Yes.

3) TRANSFER TO DISC:-
MERGE first generator program from tape
Alter loading instructions in line 1 to :- LOAD d1"spco" CODE and LOAD d1"sprbas"
SAVE d1 "sprgen" LINE 1
Enter CLEAR 25231
LOAD code block from tape
SAVE d1"spco" CODE 25232,40304
Reset Spectrum
MERGE program from tape
SAVE d1"sprbas" LINE 98
To load program from disc:- LOAD d1 "sprgen"

4) MAX SPRITE SIZE (character sqrs):-
6x6

5) MEMORY STORAGE CAPACITY:-
79 frames size 3x3. Less with larger sprites.

6) SPRITE ANIMATION TEST?:- Yes.
Number of frames and speed user-defined.

7) ADVANTAGES:-
A screen can be loaded into the designer, and sprites grabbed from or put into it. Screen can be re-saved. Sprites can be saved with frame information for re-loading into generator, or as a string of bytes. In addition to usual scroll, mirror, turn, invert, etc. facilities, there are also thicken and outline options. A chequered pattern, in character square size, using alternate BRIGHT squares, can be laid over enlarged and normal size screens. Design is repeated on normal size screen as you draw on enlarged screen.

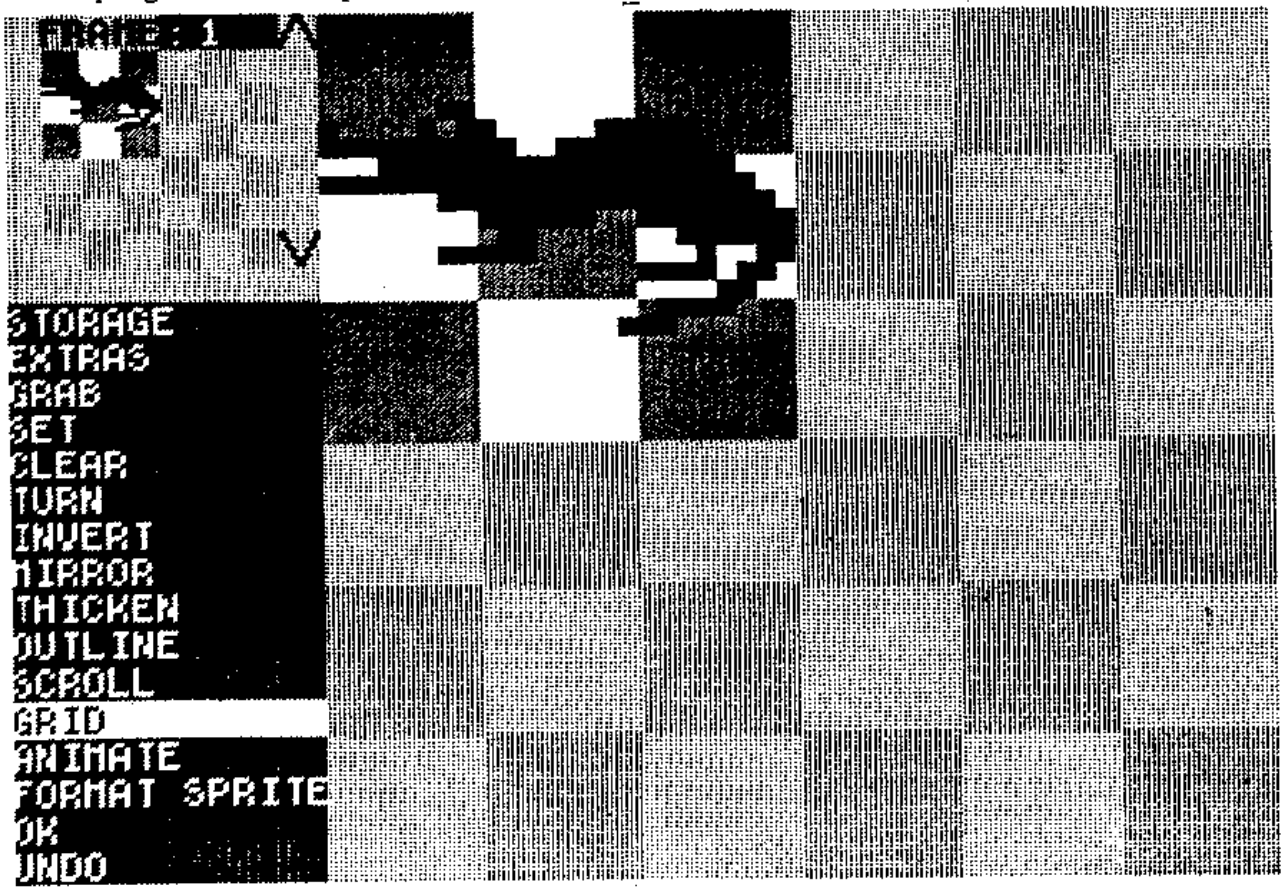


Fig 4

8) DISADVANTAGES:-
No pixel grid on enlarged screen. Sprite file memory rather limited, especially if using large sprites. Pixel setting cursor is also used to select menu options, so there is a lot of unnecessary cursor movement. Arrow keys are not used for cursor movement.

9) VERDICT:-
A versatile and sophisticated generator, marred for me because there is no pixel grid.

Fig 5

ANIMATOR-1 - Softcat (Fig.5)

1) OVERVIEW:-
A versatile sprite generator/screen design package.

2) SPRITE GENERATOR SEPARATE?:- No.
Whole package is aimed at sprite design and handling.

3) TRANSFER TO DISC:- Easy.
MERGE first program from tape
Alter load instructions in line 1 to:
LOAD D1"ANC" CODE -and- LOAD D1"ANB"
SAVE d1"Animator" LINE 1
Enter CLEAR 33963
LOAD code block from tape
SAVE d1"ANC" CODE 33964,31572
Reset Spectrum
MERGE next program from tape
SAVE d1"ANB" LINE 9999

To load from disc: LOAD d1"Animator"

4) MAX SPRITE SIZE:- Full screen

5) MEMORY STORAGE CAPACITY:-
Up to 255 frames, depending on size.

6) SPRITE ANIMATION TEST?:- Yes.
Number of frames and speed user-defined.

7) ADVANTAGES:-
All graphics package facilities, (line and circle drawing, texture fills etc.) available for sprite designing. Sprite file can be saved in character squares (for use in BASIC programs) or in lines of bytes. Optional save of attributes with sprite file. Enlarge window can be moved around screen, or switched off altogether.

8) DISADVANTAGES:-
Poor handbook. Fill operation can over-write the sprite file, though this can be avoided by using the disc/microdrive as a buffer.

9) VERDICT:-
The most versatile of all the designers, if you have the ability to design your own sprites. Not so easy to use if you are copying from a book.

Finally, for those who, like me, could not design an original sprite if their life depended on it, there are sprite libraries around. LASER BASIC comes with two useful sprite files. And, of course, several of the programs let you grab sprites from existing screens, so a screen snapshot or two from games programs can be a useful source.

There is a splendid library of over 200 sprites, some with related frames for animation, in a book:-

Step-by-step programming: ZX Spectrum and ZX Spectrum+. Book 4 - Graphics. By Piers LETCHER.
Screen Shot programming series published by Dorling Kindersley. £5.95

Sadly, this is out of print now, but you may find an odd copy in a bookseller, or your local library should be able to get hold of it for you. The sprites in Figs. 1, 3 and 4 are from this collection. They are all displayed in the book in 'life size' and on an enlarged pixel grid, and there is also a list of decimal bytes for each one.

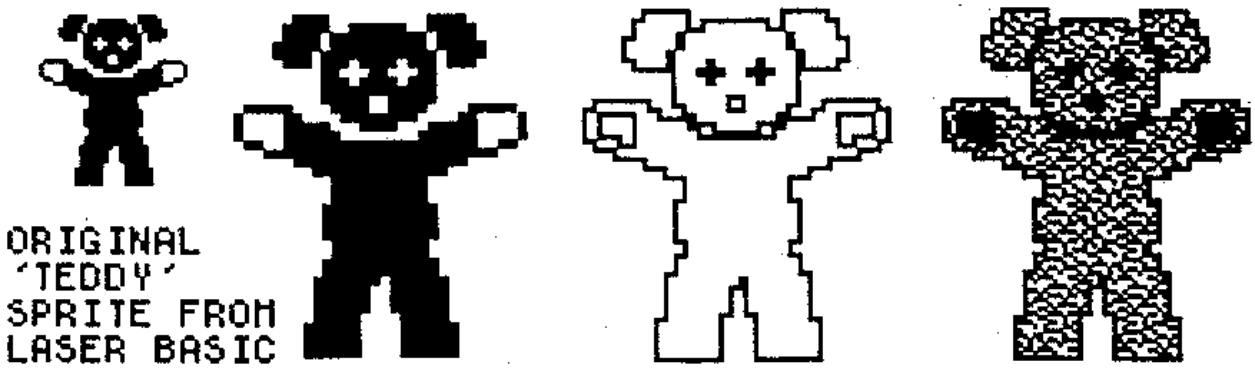


Fig 6

Don't be afraid to play around with sprites, to improve or customize them. Recently, I needed a sprite of a teddy bear for a children's poster. The one I found in the LASER BASIC file looked more like a golliwog who had become separated from his head! Fig.6 shows you how, with a few simple steps, it was transformed. And if you turn to your Feb'89 copy of FORMAT (Vol 2 No 7) and look closely at the picture of St.Cecilia in Fig.1 of my article on Artist II, you may recognise the bathing beauty from Animator 1's demo. (This month's Fig.5). Her beatification achieved by equipping her with a long dress, a halo and a church organ.

So whether or not you can draw, or program in machine code, you can still work with sprites and produce sophisticated animation, using one or other of these programs.

PROGRAM 1.

```
   2 LOAD D1 "GEN"CODE : POKE 23675,62
     : POKE 23676,194: CLEAR 49726: DI
     M S(257): DIM Z(27): DIM Y(27)
3680 REM
3715 GOTO 2030: PRINT AT 21,0;"INPUT S
     PRITE FILE (1-5 OR A-Z)"
3720 LET A$=INKEY$: IF CODE A$<48 OR C
     ODE A$>90 THEN GOTO 3720
9998 CLEAR 60000: GOSUB 91: POKE RADD+
     6,0: POKE RADD+7,0: POKE RADD+8,0
     : POKE RADD+9,0: SAVE d1"SPTGEN"
     LINE 1: SAVE d1"GEN"CODE 49898,30
     00: VERIFY d1"SPTGEN": VERIFY d1"
     GEN"CODE : STOP
```

PROGRAM 2.

```
   5 REM ATTR 0,7,1,0
  10 REM WINDOW 16,1,14,20
  15 REM CLW
  20 REM CIRCLE 184,88,40
  25 REM CIRCLE 154,50,10
  30 REM PLOT 160,120
  35 REM DRAW 157,123
  40 REM DRAW 147,113
  45 REM DRAW 146,114
  50 REM DRAW 145,120
  55 REM DRAW 145,123
  60 REM DRAW 147,129
  65 REM DRAW 152,133
  70 REM DRAW 158,135
  75 REM DRAW 165,135
  80 REM DRAW 168,134
  85 REM DRAW 157,123
  90 REM CIRCLE 170,98,10
  95 REM CIRCLE 173,95,5
 100 REM FILL 169,100,9
 105 REM PLOT 184,80
 110 REM DRAW 180,80
 115 REM DRAW 170,81
 120 REM DRAW 165,82
 125 REM DRAW 164,81
 130 REM DRAW 165,78
 135 REM DRAW 170,73
 140 REM DRAW 175,71
 145 REM DRAW 182,70
 150 REM DRAW 184,70
 155 REM FLIP2 16,1,7,20,23,1
 160 REM PLOT 170,68
 165 REM PLOT 168,65
 170 REM PLOT 184,87
 175 REM DRAW 167,65
 180 REM DRAW 168,64
 185 REM DRAW 183,87
 190 REM DRAW 185,87
 195 REM DRAW 169,64
 200 REM DRAW 184,87
 205 REM PLOT 184,88
 210 REM DRAW 184,122
 215 REM DRAW 185,122
 220 REM DRAW 185,88
 225 REM DRAW 183,88
 230 REM DRAW 183,122
 235 REM END
```

HACK-ZONE

By: Hugh J. McLenaghan.

Hello and welcome to another Hack-Zone. This month I will start with some pokes sent in by Mr P.Probert from Cumbria:-

SILKWORM

First loose a life, then do the following pokes:-

```
POKE 30780,0
POKE 39663,0
POKE 39969,0
POKE 38561,0
```

After you type these pokes and return to the game, you will be INVINCIBLE.

Mr Probert also sent in two programs which he used to find the above pokes, I have joined them together which makes it a lot faster than two separate programs.

Type in and SAVE this program:-

```
 10 CLS #
 20 PRINT TAB(10);"POKE-FINDER"
 30 PRINT TAB(9);"By P. Probert"
 40 PRINT ''
 50 PRINT "Write down all the numbers
    that"'"are printed on the screen
    ."''
 60 FOR F=24220 TO 65535
 70 IF PEEK F<>33 THEN GOTO 100
 80 IF PEEK (F+3)<>53 THEN GOTO 100
 90 PRINT "Code at ";F+3
100 IF PEEK F<>58 THEN GOTO 130
110 IF PEEK (F+3)<>61 THEN GOTO 130
120 PRINT "Code at ";F+3
130 NEXT F
140 PRINT '''FINISHED!"
```

To use the program, you must do the following:-

1) Load up the game you wish to look for the pokes in.
2) POKE 23730,250 and POKE 23731,95
3) Change PC to 4535
4) Return to game
5) Load and run the POKE-FINDER program.
6) After writing down all the numbers, you must reload the game and POKE each number with 0, one at a time, until you get INFINITE lives or something else.

Thank you Mr Probert for sending in the programs and POKES.

If you have pokes or programs which you think may be useful, then send them to me. Remember Hack Zone is not just about games. It's about all aspects of altering commercial programs.

Also if you have any ideas, problems or questions you want answered, then please write. If I do not get these things, then I am afraid that the Hack-Zone will have to be stopped, as I do not have any more ideas myself.

I need your help, so please try. Send your letters to:-

Hugh J.McLenaghan. (Hack Zone),
36, Floorsburn Crescent,
Johnstone,
Renfrewshire,
Scotland,
PA5 8PF.

EDITORS NOTE

The Hack Zone has been hard work for Hugh. He has asked (several times) before for contributions. Whats the matter with Hack Zone readers, do you want everything done for you? So search out those POKEs, list those program mods, and write to Hugh so the Hack Zone can continue. **Ed.**

DISCiPLE ANATOMY

By: Dick Guy.

This month we start a short series on the inner workings of the DISCiPLE. Before we start however I have to say the following. The circuits which will be presented should only be taken as representative of the DISCiPLE, for the following reasons:-

a, I am only human, so errors may occur. (I hope not).

b, All manufacturers reserve the right to change their designs in order to improve their product.

As the saying goes "no responsibility will be accepted. . . . . ".

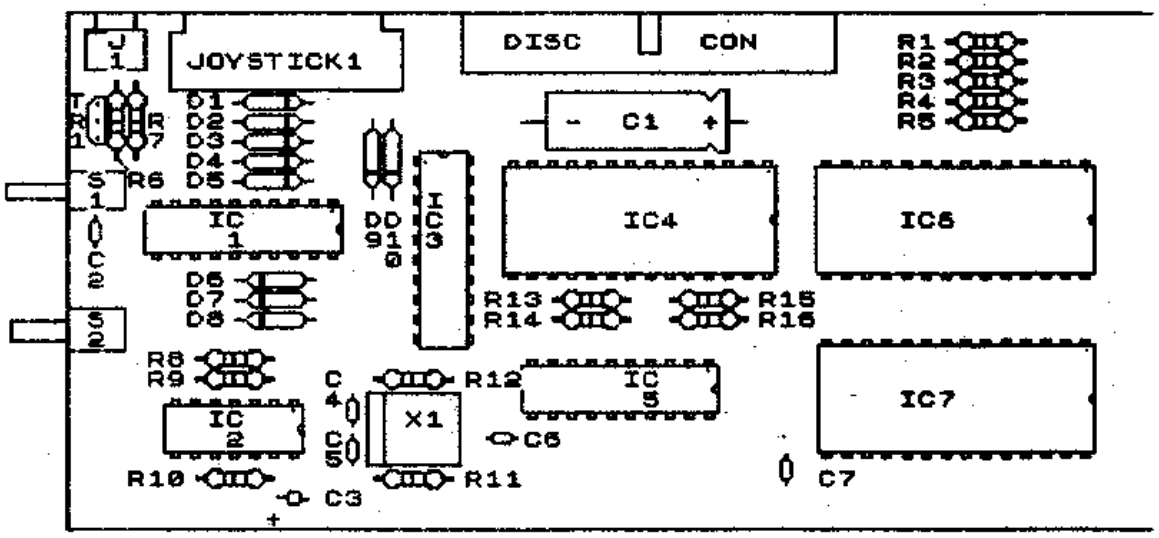
We start this month with a layout diagram of the DISCiPLE as would be seen when the top cover is removed. We start this way so the circuits to be described can easily be located. Also included is a diagram of the DISCiPLE rear edge connector. For the more observent among you, no, I haven't made a mistake there really are four more connections on the edge connector. What they can be used for will be seen when we get to them.

Before finishing for this month a brief discussion on buses. No, not the sort that are always late, but the sort to be found under the bonnet of computery hardware.

In general the internal connections of computer hardware can be allocated to one of four buses:-

a, Address bus. This contains the 16 address lines.

b, Data bus, which contains the 8 data lines.



DISCIPLE- TOP VIEW FIG 1A

S1 INHIBIT S2 SNAPSHOT

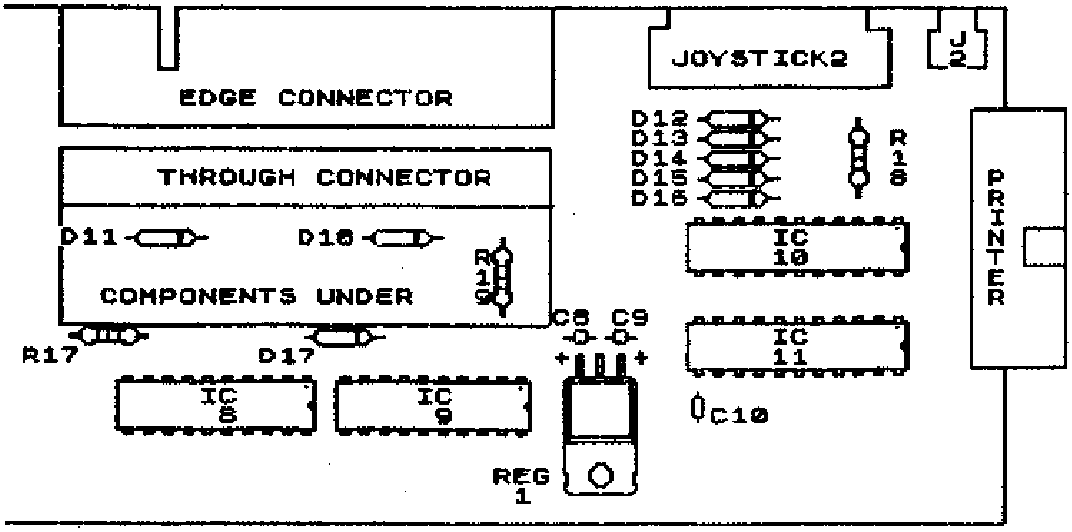
R1-5 2K2	C1 220 UF 16V	TR1 BC337
R6 3K9	C2 0.01UF	D1-10 BAW76R
R7 560	C3 22 UF 35V	
R8-9 330	C4 1 UF 35V	
R10 560	C5 33 PF	
R11-12 2K2	C6-7 0.1 UF	NOT TO SCALE
R13-16 330		

c, Control bus, which contain the standard 15 Speccy control lines, plus a few more.

d, Power supplies. Some may not agree with this concept but it may help to consider them as such.

Why, you may be asking, is he talking about buses? Simple. The circuit of the DISCiPLE is quite complex and in order to present a clear layout in FORMAT, I have chosen to break the whole thing down into manageable chunks. Evidently some means is needed to "join" them all together and that is where the buses come in.

Thats it for this month. Next month we start in earnest with the disc drive circuitry. See you then.

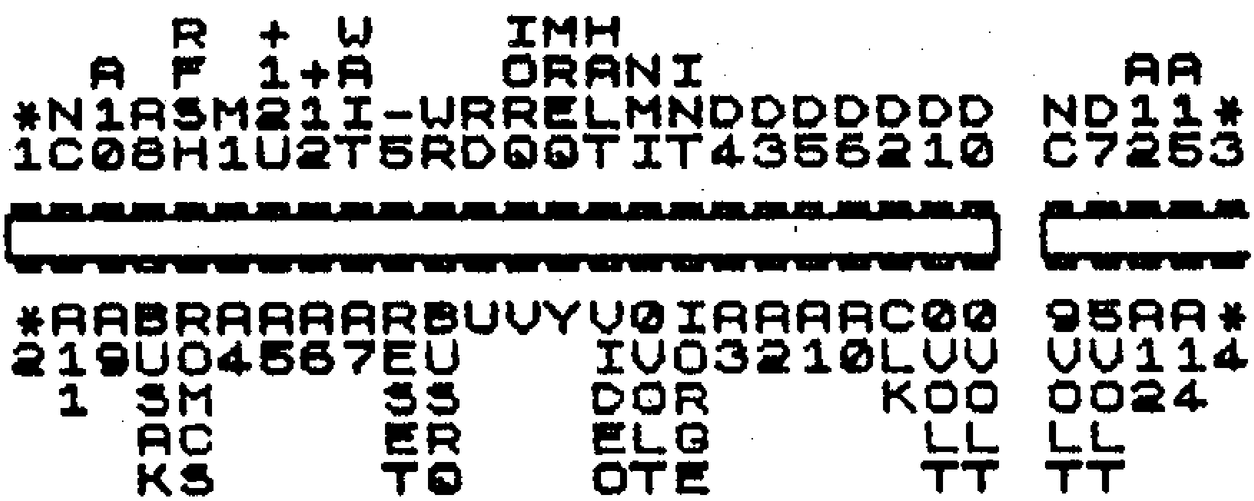


DISCIPLE-TOP VIEW FIG 18 NOT TO SCALE.

R17 330
R18 560
R19 470
D11-18 BAW 76R

REG1 LM340T
C8 22UF 35V
C9 1UF 35V
C10 0.1UF

DISCIPLE EDGECONNECTOR AS SEEN



HACKER'S WORKBENCH REVIEW

By: Dale Nanson.

I have read with great interest the many articles on hacking into programs that have been published in FORMAT over the years and have frequently wanted to try some hacking myself. Unfortunately PICK-POKE-IT and PLUS D HACKER both only work on the PLUS D. I have a DISCiPLE. When I contacted MGT about when a DISCiPLE version of PICK-POKE-IT would be available I was told 'never'. I did not give up and recently I found a program that does allow a DISCiPLE to hack into snapshot files.

This program is the 'HACKERS WORKBENCH by S.D.Software' and although the program is now nearly two years old it has never been advertised or marketed in any way. After speaking to Bob Brenchley and S.D.S. I understand it is now to be put onto the market, so I thought FORMAT readers would welcome a review.

Hackers Workbench is supplied on a cassette and comes with a small 19 page users guide. The Hackers Workbench manual does not try to teach you how to hack, there are enough books on this already, but it does contain all the info you need to start hacking. To load Hackers Workbench all you have to do is to boot your DISCiPLE and then load the cassette. The program loads, saves itself to disc, and auto runs.

You are presented with a menu of 16 options. Each option selected by a single key press and you are given on-screen prompts whatever you do. The first option I use is 'd' to select the way the program presents its printouts. I actually like working in HEX and so select the all HEX option. There are 4 in all allowing data to be printed in ASCII, HEX or Decimal and addresses to be printed in HEX or Decimal.

The next option to use would be 'a' to load a snapshot file. The file is selected by entering the P number that you get with a full CAT of the disc. (A CAT function is available in option 'b'). It is now that you find how Hackers Workbench is different from all the other hacking programs. It does NOT load the snapshot into memory. Instead it works by modifying the disc file. This means that the disc with the snapshot has to be in the drive at all times. It also means that if you get your pokes wrong you do not lose everything when the machine crashes because it is all safe on disc.

The two most useful features of Hackers Workbench I find are options 'i' to search for a data pattern and 'k' for comparing two snapshots. The find option is quite fast considering that the whole file has to be read from disc and will find all occurrences of anything from a single byte to a string of more characters than I could type in. It claims the limit is 4096 bytes!

A word here about how you enter the data. Whenever Hackers Workbench needs a number from the user, eg a memory start address, it can be entered as a decimal number or as a HEX number but the entering of HEX numbers could be rather strange if you are not used to the 'C' programming language. To enter a HEX number you have to type '0x' followed by the HEX digits. eg 255 in HEX would be entered as '0xff'. You can change between the two types at any time and Hackers Workbench works out what you mean.

If you have to enter data, e.g. when entering a pattern to find then you can enter it in any of three ways. The first is in ASCII for this you have to type the string of characters inside

quotes and you have to type the quotes. Or you can enter it as a hex string by the same method as for numbers except it can be much longer, the final method is by decimal numbers separated by commas. The example given in the manual is to enter 'Hack' you would type "Hack" or 0x4861636b or 72,97,99,107 . Once agian you type it and Hackers Workbench works it out.

The only problem I had with this was you can not use expressions such as 42+1024*32 you have to work out the answer and type 32810. This can be annoying when using the disassembler as you can not restart the disassembly at a relative address. This would be useful when you come to a JR instruction. The disassembler itself is very fast indeed and does not suffer from the annoying habit of getting lost after a RST 8 instruction that many disassemblers have.

As I said before, the compare function is very useful indeed and is a unique feature of Hackers Workbench (I'm told). With this you can find all the bytes in two snapshots that are different. So if you want to know where the number of lives left is held then just take a snapshot with 5 lives and another with 3 then compare the two snap shots. You should try to keep as much as possible the same otherwise you can get swamped with numbers but a bit of searching and you will soon find the byte that has changed from 5 to 3. It is then a simple matter to find every place where it is used in the program by using the find function. For example if you find that the number of lives left is held at memory address FE42H (or 0xFE42 if you prefer) then you do a find of 0x42FE and get a list of every place the program uses that memory location. Using this method I managed to get infinite lives, keys and ammo in Eagles Nest in under 40 minutes. And I'm only learning.

Hackers Workbench also lets you do many other things like looking at and changing the Z80 registers, moving and filling areas of the snapshot but this is all a bit beyond me.

A word of warning, when I first tried the program I wasn't using the latest level of the DISCiPLE system file and so couldn't get Hackers Workbench to work. But after a call to the Help Line to explain it to me, and typing in the updates to the file to make it Sys 3d from issues 10 and 11 of FORMAT everything worked fine.

Another annoying thing about Hackers Workbench is that if you are using a 128K snapshot and you are using addresses above 49152 it always asks you for a page number 0-7. It would have been better if you could select a page and then just use that until you changed it. In the same way when an address is printed the page number is printed out in front of it. I think this looks untidy but I am getting used to it.

As a final note I am also told that HACKERS WORKBENCH works on both DISCiPLE and PLUS D so can be used for both systems. At £9.90 its a snip, even if you already have one of the other utilities this one offers several features that make the purchase worth while.

HACKERS WORKBENCH is available from:-

S.D.Software,
16, Octavia Street,
Kirkcaldy,
Fife,
Scotland,
KY2 5HH.

Cost £9.90 plus 50p Post & Packing (£1.20 OVERSEAS).

TTX2000 REVIEWED

By: John Wase.

The TTX2000 is exactly what it says; a teletext adapter for Spectrums. Attach it to your micro as instructed, plug in the lead from a good roof aerial, tune it in and bingo; there's your pictures (or so the instructions imply). In practice, it's not quite so simple, but it's not difficult either. This is how the review tests went.

Firstly, I want you to understand that I have no teletext facility on my television, and have never used one. If I had, I suspect few difficulties would have arisen. However, let me start at the beginning. The box contains the adapter (about the size of a VTX modem - or a videotape box), a power pack, 56 way ribbon cable with three outlets and paddle board, and a twelve page instruction book. Although this is labelled 1989, the gadget was clearly designed some years ago, and this leads to some problems. Whilst it fits fine under the original rubber keyed Clivemicro, it is far less happy under the +2: I eventually found the best way was upside down behind it (and I didn't even dare try it with the +3). On powering the thing up, one immediately gets the main menu (not really a menu, just a list of prompts) - Enter jumps straight in and pressing C1 to C4 selects the channel. Then Pnnn where nnn is a 3-digit number gets you the page you need, H holds the page if the display grinds through several, R reveals hidden messages like answers to a quiz, E exits to Basic, Z copies page to printer and S saves the page to microdrive. Fine, except first you have to tune the thing in.

So, switch on to the main menu, Enter, press C, then 1 (channel 1), then C (switch off Automatic Frequency control) then turn tuning control fully clockwise with a screwdriver (until it clicks) then slowly back until a header appears with CEEFAX and page numbers mainly in the range 100 - 199. Well, that's what it says. My younger son had great difficulty in getting any channel tuned in. If you are not used to teletext you don't know which pages belong to which channel, nor do you realise that choosing a page not available on that channel merely gives a blank. The instructions could well have been more detailed at this point. In addition, they failed to tell you which pages are available on channels other than 1, and this gave us great difficulty, as it meant that we didn't know what to tune to. However, once we'd tuned in it was straightforward. Well almost.

With the RGB of the +2 (yes, it works on a +2) straight into my monitor telly, screens came up crisp and sharp. Key in P123 to select page 123 and the screen goes blank except for the page numbers churning over at the top until your selected page is accessed: the Z80 is not desperately quick at searching. Apart from this, the display was fine and we had great fun finding the current market prices of bananas and other useless information.

So essentially the thing works and works pretty well. However, as I mentioned earlier, it was clearly designed some time ago. My microdrives have long since died, so I tried connecting it to +D, Disciple and Discovery disc interfaces. With the +D, there was no picture, no little light on the +D, nothing. I deduced a major port clash. Disciple was a little bit better. You could get the menu screen, but keying Enter merely gave you part of the heading (P100) and the thing locked there with a blank screen. Usually, attempts to enter Basic resulted in a crash.

Powering up disenabled and subsequently gently pressing the enable button got Basic after the fourth or fifth attempt, but RUN merely gave OK: in other words, there was no way you could boot the system. Discovery gave Basic all right, and you could even format discs, but keying Enter from the main menu gave the screen number, just like the Disciple, and locked everything up. So again, no go. Finally, the illustration in the manual shows an old ZX printer in use, presumably for the page copy function (key Z), but as I couldn't connect any useful interfaces (and so no printer port) I didn't pursue the matter. All of this shows how important it is not to have port clashes where peripherals are concerned, and explains why MGT are allocating ports or groups of ports for their SAM, so that third party manufacturers do not run into these problems.

The best bet would be if you have upgraded (e.g to a Spectrum+, a +2, +3 or 128), and have an old rubber-key on the wardrobe. Sit the thing in the corner with a switched splitter on the telly aerial, and you're in. For at the price (£60), it's a bargain - if you don't need hardcopy from the pages.

The TTX2000 Teletext Adaptor is available from:-

Micro Projects Ltd.,
Freepost,
Alsager,
Stoke-on-Trent,
ST7 2BR.

Tel: 0270-875178.

Price:- £59.95 incl p&p.

* + * + * + *

NEV'S HELP PAGE

By: Nev Young.

The first cry for help this month resulted in me having to phone somebody else for help to answer it. The problem, that K Newby of Telford has, is trying to 'PUT' a file using Hisoft Gens 3 and keeps getting a 'Protected' message.

I don't have Gens and so know nothing about it. But when I phoned for help I got the following reply. "Get Version 4 of Devpack (the latest version) from Hisoft and state clearly that it is for use on the PLUS D when ordering". I'm afraid thats the best I can do for that one, unless some one out there knows better.

Next an easy(ish) one. "How can I do a POKE @6,0 (1) from machine code". On the +D use the following code:-

```
RST 8           ; page IN the PLUS D
DEFB 71
LD A,0          ; the value to poke
LD (8192+6),A   ; 8192+POKE address
CALL 79         ; page OUT the PLUS D
```

On the DISCiPLE (GDOS 3d or higher) replace the value 8192 with 664. For DISCiPLEs not using the current DOS use:-

```
IN A,(187)      ; page IN the DISCiPLE
LD A,0          ; the value to poke
LD (664+6),A    ; 664+the POKE address
OUT (187),A     ; page OUT the DISCiPLE
```

As you can see by paging the shadow memory in you can then read and write any bytes of the PLUS D or DISCiPLE memory.

And now another quick one from Juan Guillen Serra in Barcelona who writes "Since upgrading from 3c to 3d the automatic short CAT no longer works with the SAVE and ERASE commands. Was there something wrong with the change to the PCAT data on line 111 of issue 11 page 14?".

Well Juan, it was known that the change to the PCAT routine would stop the auto cat but as so few people used it it was thought to be an insignificant loss. The change being made to make the DISCiPLE and PLUS D work the same way for the PCAT hook code. However as you are the person who used it, and obviously want it back then simply add the basic statement **CAT *!** after the save or erase in your program. I have to admit that I was glad to get rid of the auto cat as it was, to me, a great annoyance.

Now for more disc compatibility problems. This one from S Young (no relation) of Southend-on-Sea is typical of the problems I had when I was a computer service engineer. He writes "I have a disc drive that will read and write perfectly well but no other drive can read discs written by this drive and this drive will not read discs written by other drives. What do I need to tweek to make it work".

A very good question and I feel deserves a better answer than you are going to get. There are 4 main adjustments on a disc drive that if wrong will cause the problem you have. They are:-

1. Disc speed
2. track 0 adjustment
3. radial alignment
4. circumferential alignment

Only the first of these can be done without special equipment.

To adjust the disc speed there should be a strobe disc on the back of the disc spindle. With the disc running and using a fluorescent light

or the light from a TV screen adjust the motor speed until the strobe pattern appears stationary. The motor adjustment will almost certainly be a variable resistor (pot) located on the disc drive pcb.

If the track 0 sensor is wrong then on a PLUS D or DISCiPLE the drive would probably work as the disc would be an exact number of tracks out and the dos would compensate for this. To adjust it properly you have to do adjustment 3 and then step the heads back slowly. The track 0 signal should stay high at track 1 but be low at track 0. It is adjusted by moving the track 0 sensor. If it is wrong then it would need moving by about 1/100 of an inch in one direction or the other. (Actually on some drives the signal changes between track 2 and 3 and is gated with the 0 phase from the head stepper motor so check your drive manual first).

Radial alignment. This is done when the heads are less than 1/2 track out of alignment. To do it properly you need an alignment disc available from the disc drive manufacturers and a dual beam oscilloscope. Move the heads out to the alignment track and with one trace connected to INDEX, internal sync, and the other to READ DATA loosen the stepper motor that moves the heads and twist it until the test data pattern is symmetrical. Then tighten the motor screws making sure the pattern stays symmetrical. Remember the heads would have to move less than 1/150 inch so you probably would not see any physical movement. You can try doing this by setting your Spectrum to CAT a disc constantly trapping errors and try making very small adjustments with a GOOD disc in the drive until it CATs OK. (Assuming all other adjustments are OK first). But using this method you'll have to be very lucky to get it right.

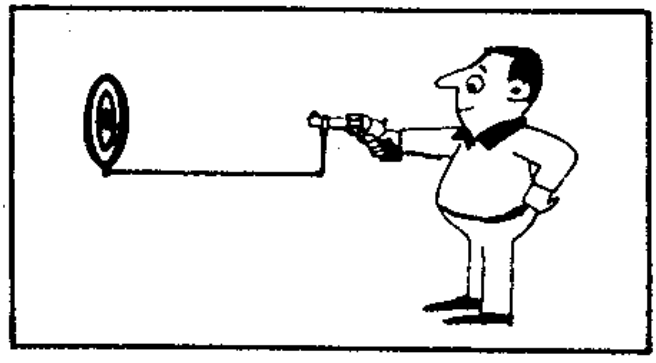
Circumferential alignment (a big name for a small adjustment) makes sure that the index sensor is correctly positioned. Using the same test setup as for 3, move the heads to the correct track and you should see a data burst at a specified time after the index mark (on my disc the data is 150-250 micro seconds after the start of the index pulse. But to be honest on soft sectored drives it doesn't seem to be too important for compatibility. The adjustment is usually by moving the sensor to get close to the time, and then there will be an electronic adjustment to do the fine tuning.

None of these should be attempted unless you have the correct jig and tools or are prepared to send the drive to a repair shop after you have botched it up. It is definitely not worth getting the tools & disc just for your own drive as they cost between £20-30 for the disc and from £500 upwards for an oscilloscope.

Also remember that once your disc is correctly adjusted it will not read ANY of the discs that it used to so copy ALL your data to tape or to another drive first.

Thats all for this month. Remember If you don't write to me I can't write this page. I also have to point out that I can not answer questions personally so DO NOT send me return postage etc. I will attempt to answer as many queries as possible but only through the magazine. Write to FORMAT or directly to me at:-

FORMAT HELP PAGE.
3, Mitchell Place,
Falkirk,
Stirlingshire,
Scotland,
FK1 5PJ.



BEYOND SIMPLE UDGs

PART 4.

By: Clyde Bish.

Last time (FORMAT Vol 2 No 10) we looked at storing pictures in memory and moving them onto the screen with a small machine code routine. Now all that's very well if you have a "128" or don't need many pics. But how can you squeeze the proverbial quart into the Spectrum's pint pot?

One way is to have a smaller "quart". In other words, perhaps you would be happy with a two colour drawing (INK and PAPER), which takes up 768 bytes less. You could go further by having only a top third, or top two thirds screen picture. (many adventures use this system, leaving the bottom of the screen clear for text). The REMs given in Program 2 last time will tell you the changes you will need to make for those variations. If you want to mix various types of illustrations in one program you'll need to POKE 65364 with the appropriate value shown in the REM statement at line 100 before you call the routine.

This technique obviously doesn't help if you really want full screen illustrations, but there is a way around that problem if you're prepared to sacrifice a little speed for a great saving in space. If you type in the following line:-

```
LOAD "" SCREEN$: FOR f=16384 TO 23295:
 PRINT #0;"Address ";f,"holds "; PEEK
f: PAUSE 20: NEXT f <ENTER>
```

after LOADing in a picture you'll see a series of numbers appear at the bottom of the screen as the program PEEKs its way through the D FILE, and later the ATTR if you wait long enough. You'll notice that the numbers 0 and 255 (and later the permanent attribute, for example 56 if the background is black on white) occur more often than any others. This is because most of the bytes of a picture are either blank (0) or inked in (255), whilst much of the attributes area remains unchanged. Knowing this it is possible to compact the data for a picture considerably by storing, for example, a line of 32 zeros as 0,32. Using this technique even a complete picture is compacted to about half its usual length. This is what the compactor routine (Program 3) does. Type it in and let's try it out.

RUN the program and answer the "attribute" prompt with attribute the illustration you're about to load in uses most of all. If you haven't the Table I suplied a couple of months ago you can calculate it yourself. Say the picture is drawn in blue ink on yellow paper the value would be 1 (ink) + 6 (paper) * 8 = 49. Now load in the SCREEN$ and wait. From now on the program takes over, and the compacting may take some time so go and make a cup of coffee, or have a stroll aound the garden. You've been hunched over the VDU for too long anyway!

When the transfer is complete the number of bytes that the picture has been compacted into is displayed. Make a careful note of this, and the title you use to save the compacted code to tape. Repeat with a new attribute value and SCREEN$ until you have all you need, then just reply to the "attribute" prompt with ENTER.

Now you have your compacted codes separately on tape you need to save them as one long code length. Do this by adding up all the code lengths you noted, and adding 67. (This is for the machine code you'll need later to "uncompact" them). Let's call the

answer T. Now **CLEAR T-1**, then load in the first compacted code to address T using:-

LOAD "title" CODE T

Load in subsequent compacted codes, adding the length of that code to the previous address, and noting the new load address. So if the first code was 2000 bytes long, the next would load in at T+2000, and so on. Hopefully, when all the codes are in you'll have 67 bytes left below the start of the udgs. Load in the data from Table B, reading across each line, with:-

FOR F=65301 TO 65367: INPUT I: POKE F, I: NEXT F

Now save the whole data and machine code block with:-

SAVE "title" CODE T,65368-T

To use this compacted code in your programs you need to have a subroutine such as:-

```
9999 RANDOMIZE A: POKE 65305,PEEK 2367
     0: POKE 65306,PEEK 23671: POKE 65
     326,C: INK C-INT (C/8)*8: PAPER I
     NT(C/8): RANDOMIZE USR 65301: RET
     URN
```

where variable A is the data start of the picture you want to call (noted when you made the one long code length) and C is the background attribute value used in the compactor program (49 in my earlier example).

Table C gives an annotated disassembly of the machine code so that readers who want smaller / two-tone / line drawings with no filled areas, (AND who understand what they are doing!) can alter the machine code to operate over less of the screen / ignore the attributes / ignore 255s and so make it run faster. (Program 3 will also need adaptation. Refer to the REMs).

In a later issue we'll be looking at making strip-cartoon-type adventure graphics, "Redhawk"-style, but before I end I'll keep the promise I gave earlier and supply a simple program of the "Kingdom" kind for those who want something to work on to use as a driver for their graphics. The listing is given in Program 4. The purpose of the game is to accumulate £100,000 by astute, if somewhat shady, trading practices. The scenario is the South China Seas, but could just as easily be smuggling along the Cornish coast, or whatever. Your ship can hold 50 units of cargo, the buying/selling price of which fluctuates. You start with £500 of your own, plus £5000 you have borrowed and must ultimately pay back. Interest is added to this whenever you change ports. Oh yes. You may run into storms en route and lose part of your cargo.

The program, which will run (badly) as listed, is in a very simple format with a simple text screen display. As listed it takes up some 3K, but this could be shortened considerably using the byte-saving tricks I demonstrated in earlier articles. I leave it to your imagination to add the scenes using picsave, compactor, or any of the other techniques I've explained earlier.

One last bit of help, though, with PRINTs and INPUTs. Printing to the screen (with speech bubbbles if you wish) is quite easy. Simply use:-

PRINT AT R,C;"text"

where R = the row, and C = the column you wish the text to appear. For inputs you'll need to use a subroutine to simulate the normal input routine, but wherever you want on the main screen. Add Program 5 to your main driver program, and set r and c to the row/column you want the input characters to appear, before you call the subroutine. Code 12 is delete (see p.183 of the old 48k Spectrum manual) so CHR$ 8 (cursor left) is used to backspace before printing the replacement character. Code 13 in line 9995 is the code for ENTER, so the subroutine returns.

Now away to the pixel paper, and get sketching!

TABLE B.

```
 33   0  64  17  80 195  26 254 255
 40  46 254   0  40  32 119  35  19
124 254  88  56 239  26 254  56  40
  9 119  35  19 124 254  91  56 243
201  19  26 213  22   0  95  25 209
 24 239  19  26 213  22   0  95  25
209  24 216  19  26  71  54 255  35
 16 251  24 206
```

TABLE C.

```
        LD HL,16384
        LD DE,start ;POKE in from basic
LOOP:   LD A,(DE)
        CP 255   ;check for filled byte
        JR Z,FILL
        CP 0  ;check for blank
        JR Z,MISS
        LD (HL),A
        INC HL
RET:    INC DE
        CP 88 ;80=2/3 or 72=1/3 screen
        JR C,LOOP
LOOP1:  LD A,(DE) ;replace with RET if
                  ;D_FILE only.
        CP attribute ;POKE from basic
        JR Z,MISS1
        LD (HL),A
        INC HL
RET1:   INC DE
        LD A,H
        CP 91
        JR C,LOOP1
        RET

MISS1:  INC DE    ;Routine to skip
        LD A,(DE) ;attributes
        PUSH DE   ;Not to be altered.
        LD D,0
        LD E,A
        ADD HL,DE
        POP DE
        JR RET1

MISS:   INC DE    ;Routine to skip
        LD A,(DE) ;D_FILE bytes
        PUSH DE   ;Not to be altered.
        JR RET

FILL:   INC DE ;Routine to fill bytes.
        LD A,(DE)
        LD B,A
BACK:   LD (HL),255
        INC HL
        DJNZ BACK
        JR RET
```

PROGRAM 3.

```
  10 INPUT "attribute value? ";c: INPU
     T "title?";a$: IF a$="" THEN STOP
  20 LOAD a$ SCREEN$
1000 LET d=16384: LET a=50000
1010 IF d>22527 THEN GOTO 4000
1020 IF PEEK d=255 THEN GOTO 2000
1025 IF PEEK d=0 THEN GOTO 3000
1030 POKE a,PEEK d: LET a=a+1: LET d=d
     +1: GOTO 1010
2000 POKE a,255: FOR f=1 TO 254: IF PE
     EK (d+f)<>255 OR d+f>22527 THEN G
     OTO 2050
2010 NEXT f
2050 POKE (a+1),f: LET d=d+f: LET a=a+
     2: GOTO 1010
3000 POKE a,0: FOR f=1 TO 254: IF PEEK
      (d+f)<>0 OR d+f>22527 THEN GOTO
     3050
3010 NEXT f
3050 POKE (a+1),f: LET d=d+f: LET a=a+
     2: GOTO 1010
4000 IF d>23259 THEN PRINT a-50000: IN
     PUT "title? ";a$: SAVE a$CODE 500
     00,a-50000: GOTO 10
4010 IF PEEK d=c THEN GOTO 5000
4020 POKE a,PEEK d: LET a=a+1: LET d=d
     +1: GOTO 4000
5000 POKE a,c: FOR f=1 TO 254: IF PEEK
      (d+f)<>c OR d+f>23295 THEN GOTO
     5050
5010 NEXT f
5050 POKE (a+1),f: LET d=d+f: LET a=a+
     2: GOTO 4000
```

PROGRAM 4.

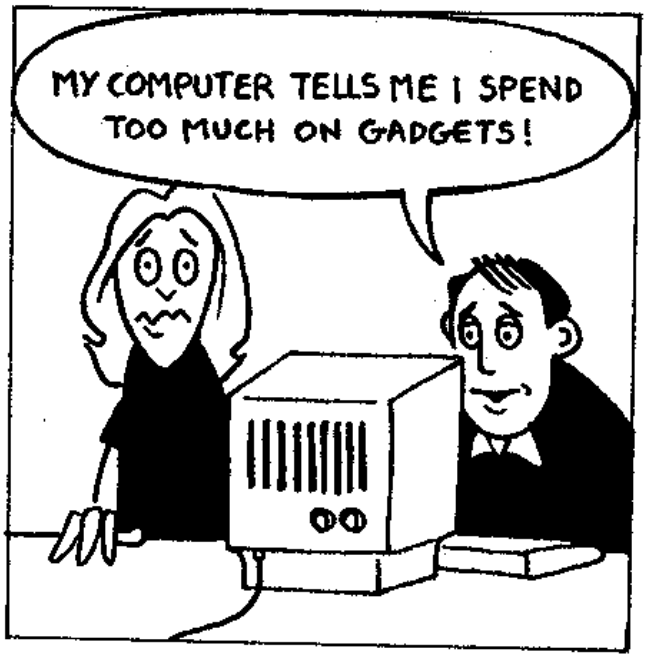
```
   1 REM From an idea by J.K.Moody
  10 POKE 23658,8: PRINT "TRADER": INP
     UT "Who is the Captain?"'n$: CLS
  37 LET s$="     ": LET t=0: LET b=500
     : LET y=5000: LET u=50: LET g=0:
     LET a=0: LET s=0: LET w=0: LET o=
     0: LET L=1: GOSUB 9900
  95 GOSUB 9800
 100 PRINT AT 0,10;n$'''"Bank
       £";b+y;s$'"Borrowed        £";y
     ;s$'"Deck Space",u;s$'"Location",
     L$+s$'"General Cargo",g;s$'"Arms"
     ,a;s$'"Silk",s;s$'"Whisky",w;s$'"
     Opium",o;s$'''"PRICES:"'"General £
     ";p(1);s$'"Arms    £";p(2);s$'"Si
     lk    £";p(3);s$'"Whisky  £";p(4)
     ;s$'"Opium   £";p(5);s$
 110 PRINT '"OPTIONS: 1.Buy 2.Sell 3.S
     ail"'("         4.Borrow" AND L=1
     )
```

```
 120 INPUT "Choose your option ";c
 170 GOTO c*1000
1000 INPUT "What do you want? ";p$
1070 LET p=(p(1) AND p$="G")+(p(2) AND
      p$="A")+(p(3) AND p$="S")+(p(4)
     AND p$="W")+(p(5) AND p$="O")
1075 PRINT #0;"You can afford ";(INT (
     (b+y)/p)): PAUSE 100: INPUT ;: IN
     PUT "How many do you want? ";q
1155 LET u=u-q: LET b=b-(q*p)
1170 IF p$="G" THEN LET g=g+q
1180 IF p$="A" THEN LET a=a+q
1190 IF p$="S" THEN LET s=s+q
1200 IF p$="W" THEN LET w=w+q
1210 IF p$="O" THEN LET o=o+q
1250 GOTO 95
2010 INPUT "What do you want to sell?
     ";p$
2020 IF RND>.7 THEN PRINT #0;"Sorry, n
     o buyers": PAUSE 100: INPUT ;: GO
     TO 100
2050 LET p=(p(1) AND p$="G")+(p(2) AND
      p$="A")+(p(3) AND p$="S")+(p(4)
     AND p$="W")+(p(5) AND p$="O")
2120 INPUT "How many to sell? ";q
2170 IF q>1 AND RND>.7 THEN LET q=q-IN
     T (RND*q): PRINT #0;"You can only
      sell ";q: PAUSE 100: INPUT ;
2180 LET b=b+p*q: LET u=u+q: IF p$="G"
      THEN LET g=g-q
2190 IF p$="A" THEN LET a=a-q
2200 IF p$="S" THEN LET s=s-q
2210 IF p$="W" THEN LET w=w-q
2220 IF p$="O" THEN LET o=o-q
2245 IF b>=100000 AND y=0 THEN GOTO 90
     00
2300 GOTO 95
3010 LET t=t+1: LET i=INT (y*.1): LET
     y=y+i: LET b=b-i
3020 INPUT AT 0,0;"Ports: 1.Hong Kong
     2.Singapore"'"3.Macao 4.Bankok 5.
     Shanghai";L
3050 GOSUB 9900: IF RND>.25 THEN GOTO
     9700
3090 GOTO 95
4020 INPUT "Borrow or Repay? ";c$
4040 IF c$="B" THEN GOTO 4500
4050 INPUT "How much? ";m
4080 LET y=y-m: LET b=b-m: GOTO 95
4500 INPUT "How much? ";m
4530 LET y=y+m: LET b=b+m: GOTO 95
9005 CLS : PRINT "You can retire with
     £";b'"You took ";t;" move";("s"
     AND t<>1): STOP
9710 PRINT #0;"STORM!": PAUSE 100: INP
     UT ;: IF RND<.5 THEN GOTO 9780
9740 PRINT #0;"Half your cargo is lost
     ": PAUSE 100: INPUT ;: LET g=INT
     (g/2): LET a=INT (a/2): LET s=INT
      (s/2): LET w=INT (w/2): LET o=IN
     T (o/2): LET u=50-(g+a+s+w+o): GO
     TO 95
9780 PRINT #0;"STORM OVER. ALL O.K.":
     PAUSE 100: INPUT ;: GOTO 95
9800 DIM p(5): LET p(1)=INT (RND*15+1)
9801 LET p(2)=10*INT (RND*14+5)
9802 LET p(3)=10*INT (RND*700+30)
9803 LET p(4)=100*INT (RND*26+5)
9804 LET p(5)=100*INT (RND*91+10)
9805 RETURN
9900 LET L$=("Hong Kong" AND L=1)+("Si
     ngapore" AND L=2)+("Macao" AND L=
     3)+("Bankok" AND L=4)+("Shanghai"
      AND L=5): RETURN
```

PROGRAM 5.

```
9990 LET a$="": PRINT AT r,c;
9991 FOR d=1 TO 20: NEXT d
9992 IF INKEY$="" THEN GOTO 9992
9993 IF CODE INKEY$=12 THEN IF LEN a$>
     0 THEN PRINT CHR$ 8;;" ";CHR$ 8;:
      LET a$=a$( TO (LEN a$-1)): GOTO
     9991
9994 IF CODE INKEY$=12 THEN GOTO 9991
9995 IF CODE INKEY$=13 THEN RETURN
9997 LET i$=INKEY$: LET a$=a$+i$: PRIN
     T i$;: BEEP .1,12: GOTO 9991
```

* - * - * - *



SMALL ADS

TWIN 5.25 40 Track single sided. Fully cased with built in power supply. Brand new. £100 o.n.o Roger Newton on 0656 720576

FOR SALE. Selection of BASIC and MACHINE CODE books for the Spectrum. Also early 'Your Spectrum' and full set of 'INPUT'. S.A.E. for full list to Malcolm Goodman, 17, Brookhill Avenue, Leeds, LS17 8QA.

SERIOUS SPECTRUM OWNERS wanting UTILITIES PROGRAMMING HELP, PRACTICAL software and USEFUL articles, GRAPHICS, INFO, IDEAS and MORE from LIKE-MINDED THINKERS, TRY ...

OUTLET monthly on MDRIVE, OPUS, DISCIPLE, PLUS D, TAPE

£2 gets YOUR FIRST issue! A blank disc or cartridge (not cassette) gets a FREE demo!

CHEZRON SOFTWARE, 605 LOUGHBOROUGH ROAD, BIRSTALL, LEICESTER LE4 4NJ

TAPESNAP will transfer snapshot files to tape. Now available as follows:- TAPESNAP 48 £4.00 TAPESNAP 128 £3.00 or both programs for £6.00 (Overseas orders add £1.00). Please send postal orders or cheques to Shimon (no mishtake) Young, 21 Colchester Road, Southend-on-Sea, Essex, SS2 6HW.

LOOKING. for PLUS D or DISCiPLE users for contact in Hungary or Austria. Istvan Ordog, Budapest, Vezer ut 143-1148.

DISC CONVERSION SERVICE. Got a game that wont snapshot? Send stamped addressed envelope for list of converted games. By Hugh (Hack Zone) McLenaghan, 36 Floorsburn Crescent, Johnstone, Renfrewshire, PA5 8PF.

YOUR ADVERT

Selling, Buying, Pen Friends, etc.

Any PRIVATE advert, up to 30 words (subject to acceptance), will be printed FREE in the next available issue. Any software sold must be original copies, in working order and with full instructions. The publishers will not be held, in any way, responsible for adverts in this column. Trade advertisers should contact the publisher for rates. If your advert remains unpublished after two issues please send it again.

* - * - * - *

BACK ISSUES

For members who have missed past issues of FORMAT (or perhaps worn theirs out through constant use) we run a back-issue service. The cost is £1 per issue (£1.25 overseas) incl p&p. Your copies will be sent out with your next monthly issue of FORMAT (provided we receive your order at least a week before). Make cheques (drawn on UK bank or Euro-Cheques, P.O., cash) payable to **FORMAT.**

AVAILABLE ISSUES

Volume 1

Issues #1 (Aug'87) - #12 (Jul'88).

Volume 2

Issues #1 (Aug'88) - #12 (Aug'89).

Please **WRITE YOUR ORDER ON A SEPARATE PIECE OF PAPER.** and mark envelope BACK-ISSUES. DO NOT include letters with your order as this will cause delays. Remember to quote your membership number or we wont be able to send you your order.

GET THE MOST FROM YOUR DRIVE FACILITIES. *STEP UP TO*

NOT SO SOFT-WARE

SPECIFICALLY WRITTEN FOR PLUS D & DISCIPLE

· SPECIFICALLY WRITTEN FOR PLUS D AND DISCIPLE · SPECIFICALLY WRITTEN FOR PLUS D AND DISCIPLE

FASTWORD PLUS

Here is a great programme that rejuvenates an old favourite. How often when using TASWORD TWO, have you been stuck for the right word? FASTWORD PLUS offers a complete reference book of words, waiting, ready for use at the press of a key!

This is the Plus D and Disciple Disc version of the sucessful program FASTWORD, a THESAURUS for the Spectrum. Features include ● Full alphabetical index of over 6,000 words ● A display of almost 600 screens ● Equivalent to a 120,000 word printed Thesaurus ● Sense identifaction for each word ● Select a word from the index using the first 2 letters and displayed on the screen is an average of 20 synonyms and associated words ● 2 way scrolling index ● Fast response ● Overlays Tasword 2 ● Words can be looked up in the Thesaurus as you are working on a text file in Tasword 2!

Also supplied is a NEW TASWORD 2 BASIC PROGRAM, specially written for Disc drive operation, offering new features such as selective catalogue, Wordcount etc. A CONVERTER program tailors your Tasword 2 code to give any combination of screen colours plus a new highly readable Tasword 2 character set.

Complete with instructions these 4 programs represent amazing value. Supplied on Disc. **ONLY £9.95**

FILE MASTER

At last! A comprehensive DataBase system that is **VERY EASY** to use. You will never forget the correct keys to press when using **FILE MASTER,** we guarantee it!

This versatile and powerful program can be used to store virtually any type of data. File Master enables you to create and maintain sophisticated data files, the type normally associated with more expensive computers. For example a file called 'DIARY' could be created. After typing the relevant data, the program can inform you which of your friends have a birthday in the next month, if the TV licence is due and the date of your next dental appointment. Records of money paid into the bank and cheques drawn could also be kept in the diary, and for an up-to-date total you simply press a key! The variety and scope of the files you can create is almost limitless and each file you create can have an infinite number of records (using segmented files).

Send for your **FILE MASTER** disc today. **ONLY £8.95**

THE ORGANISER

Features include ● Organisation of programs on any disc in any order you wish ● Makes finding programs in the directory much easier ● Selected programs load faster ● Checks for faulty sectors ● Automatic or manual Sector Repair ● Recovers erased files ... PLUS many more features to make file handling and disc 'housekeeping' so much EASIER.

This is what the experts said:-

"The ORGANISER has a very friendly user interface.... it's rare to find a package so easy to use.... brings out the best features in GDOS.... MGT should bundle it with SAM.... value for money...."
SID MARTIN in COMPUTER SHOPPER.

"Beautifully constructed and presented.... at £5 a bargain.... worth £10 or more just for the Disc Doctor aspects alone.... clearly to be recommended...."
Dr. JOHN WASE in POPULAR COMPUTING WEEKLY.

The ORGANISER is an essential program for ALL Plus D and Disciple users. Supplied on Cassette. **ONLY £5**

DISC MANAGER

THE DISC MANAGER is the most powerful program ever written for the DISCIPLE/PLUS D.

Designed to take advantage of Disc Drive ownership, the Manager keeps track of all the programs on all your discs, ● Storage of up to 27,000 records on one Disc, or 79,920 total. Random File Access. ● Name & Number Discs with fast Autonumber and user pre-defined titles features. No typing in of Data. Press a key and Discs are automatically added to appropriate catalogue. ● Fastest ever M/Code Search. 2 modes – Search and Load or Search and List all occurances, then select program to load. ● Plus many other unique features. ● Comes with 16 page manual and demonstration catalogues. ● Operates with 48K or 128K Spectrums. **Send for the DISC MANAGER today... and you'll soon wonder how you ever Managed without it!**

SPECIAL INDUG PRICE

NORMALLY £14.95 £12.95

BETTERBYTES

SPECIAL OFFER! Send us a disc (with system), the return postage and £1 (deductable if you order FMaster) and we will send you a **USABLE FILE MASTER DEMONSTRATION** + Demo files

10 SPITTAL TERRACE · GOSFORTH
NEWCASTLE UPON TYNE NE3 1UT · TEL: (091) 285 6185

IF OUTSIDE EUROPE SEND £1.50 p&p FOR EACH ORDER

SOFTWARE THAT IS SPECIFICALLY WRITTEN FOR PLUS D & DISCIPLE

PLEASE STATE YOUR DRIVE TYPE, SIZE, ETC, INDUG MEMBERSHIP No. AND IF REQUIRED FOR DISCIPLE OR PLUS D